IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

VONAGE HOLDINGS CORP.,	)	
	)	
Plaintiff,	)	C.A. No. 07-507-***(LPS)
	)	
v.	)	JURY TRIAL DEMANDED
	)	
NORTEL NETWORKS, INC. and	)	
NORTEL NETWORKS, LTD.,	)	
	)	
Defendants.	)	

NORTEL'S ANSWER AND COUNTERCLAIMS

Declaratory judgment defendants Nortel Networks Inc. and Nortel Networks Ltd. (collectively "Nortel"), by their attorneys, answer the Complaint for Declaratory Judgment of plaintiff Vonage Holdings, Corp. ("Vonage") as follows:

THE PARTIES

1. Nortel is without information or knowledge sufficient to form a belief as to the truth of the allegations of Paragraph 1 of the Complaint and therefore denies the same.

2. Nortel states that Nortel Networks Inc.'s principal place of business is located in Richardson, Texas.

3. Admitted.

4. Admitted.

NATURE AND BASIS OF ACTION

5. Admitted.

JURISDICTION AND VENUE

6. Admitted.

7. Admitted.

THE PATENTS-IN-SUIT

8. Defendants admit that Nortel is the owner of U.S. Patent No. 6,091,808 ("'808 Patent") and that a purported copy of the '808 Patent was attached as an exhibit to the Complaint.

9. Defendants admit that Nortel is the owner of U.S. Patent No. 6,445,695 ("'695 Patent") and that a purported copy of the '695 Patent was attached as an exhibit to the Complaint.

10. Defendants admit that Nortel is the owner of U.S. Patent No. 7,050,861 ("'861 Patent") and that a purported copy of the '861 Patent was attached as an exhibit to the Complaint.

BACKGROUND FACTS

11. Admitted.

12. Admitted.

13. Admitted.

14. Defendants deny the allegations of Paragraph 14 of the Complaint except to admit that Vonage has asserted, in Civil Action No. 4:04-cv-00548, currently pending in the Northern District of Texas ("Texas Action"), that defendant Nortel Networks Inc. infringes U.S. Patents Nos. 4,782,485; 5,018,136; and 5,444,707.

15. Defendants deny the allegations of Paragraph 15 of the Complaint except to admit that in the Texas Action, defendant Nortel Networks Inc. has requested a declaratory judgment adjudicating its rights with respect to U.S. Patents Nos. 4,782,485; 5,018,136; and 5,444,707.

16. Defendants admit that defendant Nortel Networks Inc. moved for leave to amend its pleadings in the Texas Action to assert the '808, '695 and '861 Patents against Vonage and that Nortel Networks Ltd. was not a party to the Texas Action. Defendants also admit that

Nortel Networks Ltd. is listed as the assignee of the patents. Defendants deny the remaining allegations of Paragraph 16.

17. Defendants deny the allegations of Paragraph 17 of the Complaint except to admit that Vonage opposed the portion of Nortel's motion to amend its pleadings that sought to assert the '808, '695 and '861 Patents against Vonage in the Texas Action.

NORTEL'S THREATS OF INFRINGEMENT

18. Defendants deny the allegations of Paragraph 18 of the Complaint except to admit that Nortel has put Vonage on notice that it believes that Vonage has infringed Nortel's patents.

19. The allegations of Paragraph 19 of the Complaint state legal conclusions to which no responsive pleading is required. To the extent a responsive pleading is deemed to be required, Defendants do not contest that this Court has jurisdiction over this matter as currently constituted.

COUNT I: NON-INFRINGEMENT OF THE '808 PATENT

20. Defendants repeat and incorporate by reference their answers to paragraphs 1-19 of the Complaint.

21. Admitted.

22. Denied.

23. Denied.

COUNT II: INVALIDITY OF THE '808 PATENT

24. Defendants repeat and incorporate by reference their answers to paragraphs 1-23 of the Complaint.

25. Denied.

26. Denied.

COUNT III:
UNENFORCEABILITY OF THE '808 PATENT

27. Defendants repeat and incorporate by reference their answers to paragraphs 1-26 of the Complaint.

28. Denied.

29. Denied.

COUNT IV:
NON-INFRINGEMENT OF THE '695 PATENT

30. Defendants repeat and incorporate by reference their answers to paragraphs 1-29 of the Complaint.

31. Admitted.

32. Denied.

33. Denied.

COUNT V:
INVALIDITY OF THE '695 PATENT

34. Defendants repeat and incorporate by reference their answers to paragraphs 1-33 of the Complaint.

35. Denied.

36. Denied.

COUNT VI:
UNENFORCEABILITY OF THE '695 PATENT

37. Defendants repeat and incorporate by reference their answers to paragraphs 1-36 of the Complaint.

38. Denied.

39. Denied.

COUNT VII:
NON-INFRINGEMENT OF THE '861 PATENT

40. Defendants repeat and incorporate by reference their answers to paragraphs 1-39 of the Complaint.

41. Admitted.

42. Denied.

43. Denied.

COUNT VIII:
INVALIDITY OF THE '861 PATENT

44. Defendants repeat and incorporate by reference their answers to paragraphs 1-43 of the Complaint.

45. Denied.

46. Denied.

COUNT IX:
UNENFORCEABILITY OF THE '861 PATENT

47. Defendants repeat and incorporate by reference their answers to paragraphs 1-46 of the Complaint.

48. Denied.

49. Denied.

AFFIRMATIVE DEFENSES

50. Further answering the Complaint, Defendants assert the following defenses in response to the allegations of the Complaint, undertaking the burden of proof only as to those defenses required by law, regardless of how such defenses are denominated herein. Defendants reserve the right to amend their answer with additional defenses as further information is obtained.

FIRST AFFIRMATIVE DEFENSE

51. With respect to each and every purported claim for relief alleged in the Complaint, Plaintiff fails to state a claim against Nortel upon which relief may be granted.

SECOND AFFIRMATIVE DEFENSE

52. Defendants reserve all affirmative defenses that are now or may become available in the future based upon any discovery or any other factual investigation.

COUNTERCLAIMS

For its counterclaims against Vonage Holdings Corp. ("Vonage"), Nortel Networks Inc. and Nortel Networks Ltd. (collectively "Nortel") states as follows:

PARTIES

1. Nortel Networks Inc. is a Delaware corporation with its principal place of business in Richardson, Texas.

2. Nortel Networks Ltd. is a Canadian corporation with a principal place of business in Toronto, Canada.

3. Upon information and belief, Vonage is a Delaware corporation with a principal place of business at 23 Main Street, Holmdel, New Jersey 07733.

4. Upon information and belief, Vonage provides broadband telephone services that utilize Voice over Internet Protocol ("VoIP") technology, which enables voice communications and telephony services over the Internet through the conversion and compression of voice signals into data packets. In order to provide these services to its customers, Vonage makes, uses, offers for sale, sells and/or imports VoIP services, equipment and products.

JURISDICTION AND VENUE

5. This counterclaim arises under the Patent Act of 1952, 35 U.S.C. §§ 1 et seq. This Court has subject matter jurisdiction to hear this action under 28 U.S.C. §§ 1331, 1338(a).

6. Venue is proper in this judicial district under 28 U.S.C. §§ 1391(b) and 1391(c) and 1400(b).

THE NORTEL ASSERTED PATENTS

7. Nortel is a technology leader and was one of the key innovators in developing many of the designs, architectures, services and products that are utilized in today's VoIP networks. Nortel owns more than 3700 United States patents, many of which are applicable to telecommunications networks, including VoIP networks. Vonage's systems, architectures, and offered services utilize technology developed and patented by Nortel.

8. U.S. Patent No. 6,091,808 ("the '808 Patent"), entitled "Methods of and Apparatus for Providing Telephone Call Control and Information," was issued on July 18, 2000. A copy of the '808 Patent is attached hereto as Exhibit A.

9. U.S. Patent No. 6,445,695 ("the '695 Patent"), entitled "System and Method for Supporting Communications Services on Behalf of a Communications Device Which Cannot Provide Those Services Itself," was issued on September 3, 2002. A copy of the '695 Patent is attached hereto as Exhibit B.

10. U.S. Patent No. 7,050,861 ("the '861 Patent"), entitled "Controlling a Destination Terminal From an Originating Terminal," was issued on May 23, 2006. A copy of the '861 Patent is attached hereto as Exhibit C.

11. U.S. Patent No. 6,937,572 ("the '572 Patent"), entitled "Call Trace On A Packet Switched Network" was issued on August 30, 2005. A copy of the '572 Patent is attached hereto as Exhibit D.

12. U.S. Patent No. 7,177,399 ("the ’399 Patent"), entitled "Determining A Geographical Location From Which An Emergency Call Originates In A Packet-Based Communications Network" was issued on February 13, 2007. A copy of the '399 Patent is attached hereto as Exhibit E.

13. U.S. Patent No. 6,823,370 ("the ’370 Patent"), entitled "System And Method For Retrieving Select Web Content" was issued November 23, 2004. A copy of the '370 Patent is attached hereto as Exhibit F.

14. U.S. Patent No. 7,035,390 ("the ’390 Patent"), entitled "User Controlled Call Routing For Multiple Telephone Devices" was issued April 25, 2006. A copy of the '390 Patent is attached hereto as Exhibit G.

15. U.S. Patent No. 6,934,279 ("the ’279 Patent"), entitled "Controlling Voice Communications Over A Data Network" was issued on August 23, 2005. A copy of the '279 Patent is attached hereto as Exhibit H.

16. U.S. Patent No. 6,337,858 ("the ’858 Patent"), entitled "Method And Apparatus For Originating Voice Calls From A Data Network" was issued on January 8, 2002. A copy of the '858 Patent is attached hereto as Exhibit I.

17. U.S. Patent No. 6,798,786 ("the ’786 Patent"), entitled "Managing Calls Over a Data Network" was issued on September 28, 2004. A copy of the ’786 Patent is attached hereto as Exhibit J.

18. U.S. Patent No. 5,991,389 ("the ’389 Patent"), entitled "Programmable Service Architecture for Call Control Processing" was issued on November 23, 1999. A copy of the ’389 Patent is attached hereto as Exhibit K.

19. U.S. Patent No. 6,799,210 ("the '210 Patent"), entitled "Dynamic Association of Endpoints to Media Gateway Controllers" was issued on September 28, 2004. A copy of the '210 Patent is attached hereto as Exhibit L.

20. Nortel is the owner of all rights, title and interest in and to the '808 Patent, '695 Patent, '861 Patent, '572 Patent, '399 Patent, '370 Patent, '390 Patent, '279 Patent, '858 Patent, '786 Patent, '389 Patent and '210 Patent (collectively "Nortel Asserted Patents") and is entitled to sue for past and future infringement.

21. The above-referenced Nortel Asserted Patents, infringed by Vonage, span key technology areas needed for modern VoIP telephony. The '808, '279, '858 and '390 Patents broadly relate to systems, devices and methods that allow users to control telephone calls and features through a computer interface (usually through a web browser). The '695, '389, '210 and '786 Patents generally relate to the management or architecture of system resources. Lastly, the '861, '572, '399 and '370 Patents broadly relate to the use of system messaging to deliver or route information to a telephone user.

CLAIM 1:
INFRINGEMENT OF U.S. PATENT NO. 6,091,808

22. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

23. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the '808 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others' efforts to do so, products, components of products, and/or methods covered by one or more claims of the '808 Patent.

24. Vonage's continued infringement of the '808 Patent is willful.

25. Vonage's infringement of the '808 Patent has damaged and will continue to damage Nortel.

26. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 2: INFRINGEMENT OF U.S. PATENT NO. 6,445,695

27. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

28. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the '695 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others' efforts to do so, products, components of products, and/or methods covered by one or more claims of the '695 Patent.

29. Vonage's continued infringement of the '695 Patent is willful.

30. Vonage's infringement of the '695 Patent has damaged and will continue to damage Nortel.

31. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 3: INFRINGEMENT OF U.S. PATENT NO. 7,050,861

32. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

33. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the '861 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others' efforts to do so, products, components of products, and/or methods covered by one or more claims of the '861 Patent.

34. Vonage's continued infringement of the '861 Patent is willful.

35. Vonage's infringement of the '861 Patent has damaged and will continue to damage Nortel.

36. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 4:
INFRINGEMENT OF U.S. PATENT NO. 6,937,572

37. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

38. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the '572 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others' efforts to do so, products, components of products, and/or methods covered by one or more claims of the '572 Patent.

39. Vonage's continued infringement of the '572 Patent is willful.

40. Vonage's infringement of the '572 Patent has damaged and will continue to damage Nortel.

41. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 5:
INFRINGEMENT OF U.S. PATENT NO. 7,177,399

42. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

43. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the '399 Patent by making, using, selling, offering for sale, or importing

into the United States, or by inducing or contributing to others' efforts to do so, products, components of products, and/or methods covered by one or more claims of the '399 Patent.

44. Vonage's continued infringement of the '399 Patent is willful.

45. Vonage's infringement of the '399 Patent has damaged and will continue to damage Nortel.

46. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 6:
<u>INFRINGEMENT OF U.S. PATENT NO. 6,823,370</u>

47. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

48. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the '370 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others' efforts to do so, products, components of products, and/or methods covered by one or more claims of the '370 Patent.

49. Vonage's continued infringement of the '370 Patent is willful.

50. Vonage's infringement of the '370 Patent has damaged and will continue to damage Nortel.

51. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 7
<u>INFRINGEMENT OF U.S. PATENT NO. 7,035,390</u>

52. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

53. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the ’390 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others’ efforts to do so, products, components of products, and/or methods covered by one or more claims of the ’390 Patent.

54. Vonage’s continued infringement of the ’390 Patent is willful.

55. Vonage’s infringement of the ’390 Patent has damaged and will continue to damage Nortel.

56. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 8:
INFRINGEMENT OF U.S. PATENT NO. 6,934,279

57. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

58. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the ’279 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others’ efforts to do so, products, components of products, and/or methods covered by one or more claims of the ’279 Patent.

59. Vonage’s continued infringement of the ’279 Patent is willful.

60. Vonage’s infringement of the ’279 Patent has damaged and will continue to damage Nortel.

61. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 9:
INFRINGEMENT OF U.S. PATENT NO. 6,337,858

62. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

63. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the ’858 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others’ efforts to do so, products, components of products, and/or methods covered by one or more claims of the ’858 Patent.

64. Vonage’s continued infringement of the ’858 Patent is willful.

65. Vonage’s infringement of the ’858 Patent has damaged and will continue to damage Nortel.

66. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 10:
INFRINGEMENT OF U.S. PATENT NO. 6,798,786

67. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

68. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the ’786 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others’ efforts to do so, products, components of products, and/or methods covered by one or more claims of the ’786 Patent.

69. Vonage’s continued infringement of the ’786 Patent is willful.

70. Vonage’s infringement of the ’786 Patent has damaged and will continue to damage Nortel.

71. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 11:
INFRINGEMENT OF U.S. PATENT NO. 5,991,389

72. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

73. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the ’389 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others’ efforts to do so, products, components of products, and/or methods covered by one or more claims of the ’389 Patent.

74. Vonage’s continued infringement of the ’389 Patent is willful.

75. Vonage’s infringement of the ’389 Patent has damaged and will continue to damage Nortel.

76. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

CLAIM 12
INFRINGEMENT OF U.S. PATENT NO. 6,799,210

77. Nortel repeats and realleges the allegations in paragraphs 1-21 of the counter-counterclaim as though fully set forth herein.

78. Vonage has and continues to infringe, contribute to the infringement of, and/or induce infringement of the ’210 Patent by making, using, selling, offering for sale, or importing into the United States, or by inducing or contributing to others’ efforts to do so, products, components of products, and/or methods covered by one or more claims of the ’210 Patent.

79. Vonage’s continued infringement of the ’210 Patent is willful.

80. Vonage's infringement of the '210 Patent has damaged and will continue to damage Nortel.

81. Nortel has suffered and continues to suffer irreparable harm and will continue to do so unless Vonage is enjoined from further acts of infringement by this Court.

JURY DEMAND

Nortel demands a trial by jury as to all matters triable to a jury.

PRAYER

WHEREFORE, Nortel prays for judgment as follows:

A. That Vonage shall take nothing by way of its claims and be granted no relief whatsoever under any of its claims.

B. That all damages, injunctive relief, costs, expenses, attorney fees, or other relief sought by Vonage be denied.

C. Judgment be entered that Vonage infringes the Nortel Asserted Patents;

D. Judgment be entered that Vonage's infringement of the Nortel Asserted Patents is willful;

E. An injunction against further infringement of the Nortel Asserted Patents by Vonage.

F. Judgment be entered awarding Nortel damages in an amount adequate to compensate Nortel for Vonage's infringement, but in no event less than a reasonable royalty rate under 35 U.S.C. § 284 and, to adequately compensate Nortel for the infringement, an accounting;

G. Judgment be entered awarding Nortel enhanced damages by reason of Vonage's willful infringement of the Nortel Asserted Patents pursuant to 35 U.S.C. § 284;

H. Judgment be entered awarding Nortel its attorneys' fees and expenses under 35 U.S.C. §285;

I. Judgment be entered awarding Nortel its costs; and

J. Judgment be entered that Nortel be awarded such other and additional relief as the Court deems just and proper.

Date: December 14, 2007

EDWARDS ANGELL PALMER & DODGE LLP

/s/ Denise Seastone Kraft

Denise Seastone Kraft (Atty No. 2778)
919 North Market Street
Suite 1500
Wilmington, DE 19801
(302) 777-7770
(888) 325-9741 (fax)
dkraft@eapdlaw.com

Counsel for Defendants Nortel Networks Inc and Nortel Networks Ltd.

OF COUNSEL:

Michael Brody
WINSTON & STRAWN LLP
35 W. Wacker Drive
Chicago, IL 60601
(312) 558-5600

Scott Samay
Howard I. Shin
WINSTON & STRAWN LLP
200 Park Avenue
New York, NY 10166
(212) 294-6700

EXHIBIT A

US006091808A

United States Patent [19]

Wood et al.

[11] Patent Number: 6,091,808

[45] Date of Patent: Jul. 18, 2000

[54] **METHODS OF AND APPARATUS FOR PROVIDING TELEPHONE CALL CONTROL AND INFORMATION**

[75] Inventors: **Timothy John Wood**; **John C. Anderson**, both of Nepean; **Shirley-Ann Milaknis**, Kanata, all of Canada

[73] Assignee: **Nortel Networks Corporation**, Montreal, Canada

[21] Appl. No.: **08/730,856**

[22] Filed: **Oct. 17, 1996**

[51] **Int. Cl.**[7] **H04M 3/42**

[52] **U.S. Cl.** **379/201**; 379/93.23; 379/216; 379/242; 379/355; 370/352

[58] **Field of Search** 379/201, 209, 379/215, 265, 93.35, 216, 202, 203, 204, 205, 93.24, 242, 93.23; 370/352, 355

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,866,758	9/1989	Heinzelmann	379/94
5,400,327	3/1995	Dezonno	370/62
5,500,891	3/1996	Harrington et al.	379/265
5,559,875	9/1996	Bieselin et al.	379/202
5,559,876	9/1996	Alperovich	379/205
5,568,540	10/1996	Greco et al.	379/89
5,590,128	12/1996	Maloney et al.	379/202
5,619,555	4/1997	Fenton et al.	379/202
5,625,682	4/1997	Gray et al.	379/209
5,661,790	8/1997	Hsu	379/230
5,689,553	11/1997	Ahuja et al.	379/205
5,742,674	4/1998	Jain et al.	379/209
5,768,359	6/1998	DiPierro, Jr. et al.	379/216
5,784,438	7/1998	Martinez	379/209
5,805,587	9/1998	Norris et al.	379/230
5,809,128	9/1998	McMullin	379/215
5,812,652	9/1998	Jodoin et al.	379/202
5,812,653	9/1998	Jodoin et al.	379/202
5,845,267	12/1998	Ronen	705/40
5,850,433	12/1998	Rondeau	379/88.17
5,884,032	3/1999	Bateman et al.	379/216

FOREIGN PATENT DOCUMENTS

96/31079 10/1996 WIPO .

OTHER PUBLICATIONS

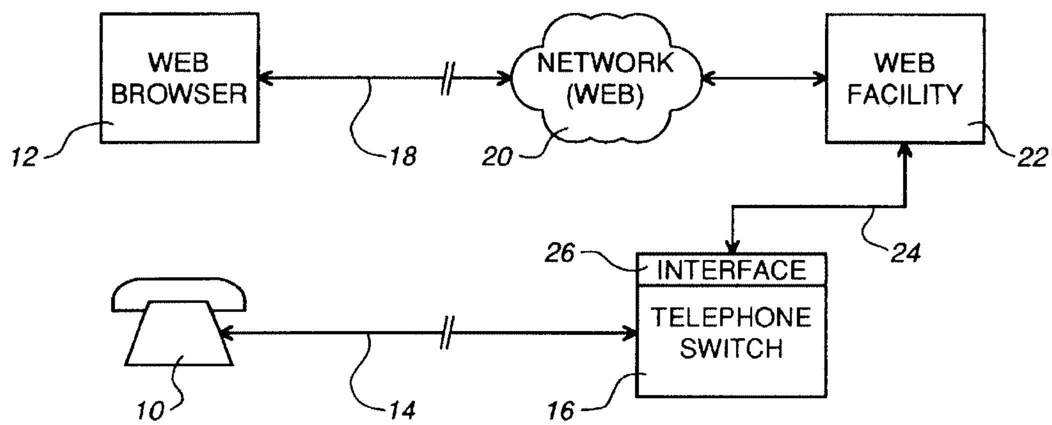
"Computer Telephone Integration: Call Control vs. Voice Processing", R. Lee, Novell, Inc., Feb. 1995.

Primary Examiner—Scott Wolinsky
Attorney, Agent, or Firm—R. John Haley

[57] **ABSTRACT**

Telephone call management is provided via a computer network (web) facility which can be remotely accessed by subscribers using web browsers. The web facility includes an information database for storing personal telephone directories and call logs, and a telephone call control system coupled to a telephone switch via a switch-computer interface. Information on calls to and/or from telephone numbers of subscribers is communicated from the switch to the web facility to be stored in the database without requiring the subscribers' browsers to be active. Subscribers can make telephone calls and control telephone communications via the browsers and the web facility. Subscribers do not require any hardware or software in addition to a telephone and web browser.

14 Claims, 2 Drawing Sheets



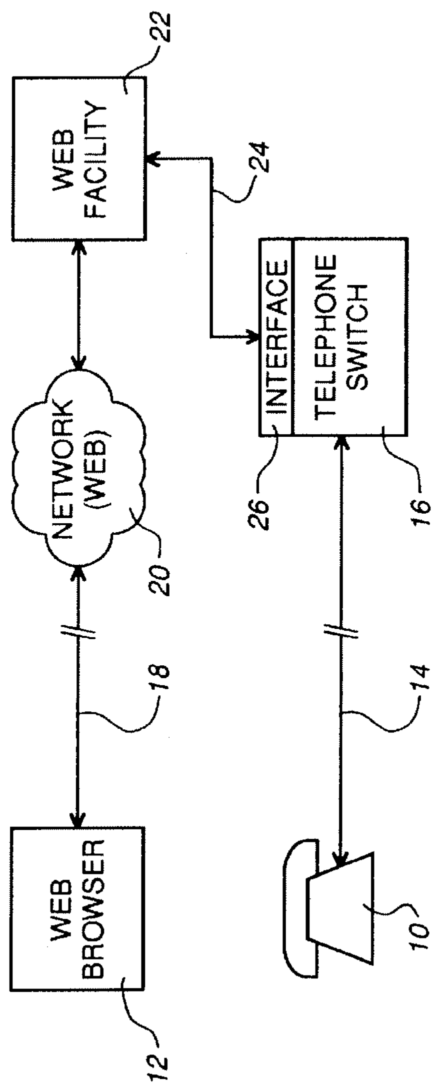

WEB BROWSER
12
18
NETWORK (WEB)
20
WEB FACILITY
22
24
INTERFACE
TELEPHONE SWITCH
26
16
14
10

Fig. 1

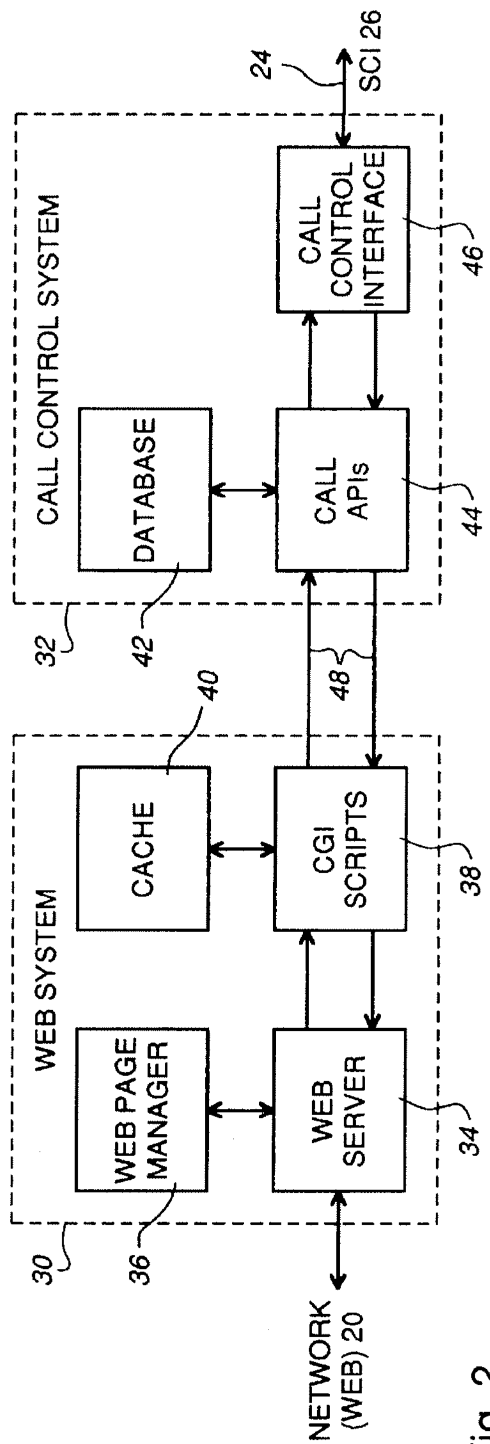

WEB SYSTEM
30
WEB PAGE MANAGER
36
WEB SERVER
34
CACHE
40
CGI SCRIPTS
38
48
CALL CONTROL SYSTEM
32
DATABASE
42
CALL APIs
44
CALL CONTROL INTERFACE
46
24
SCI 26
NETWORK (WEB) 20

Fig. 2

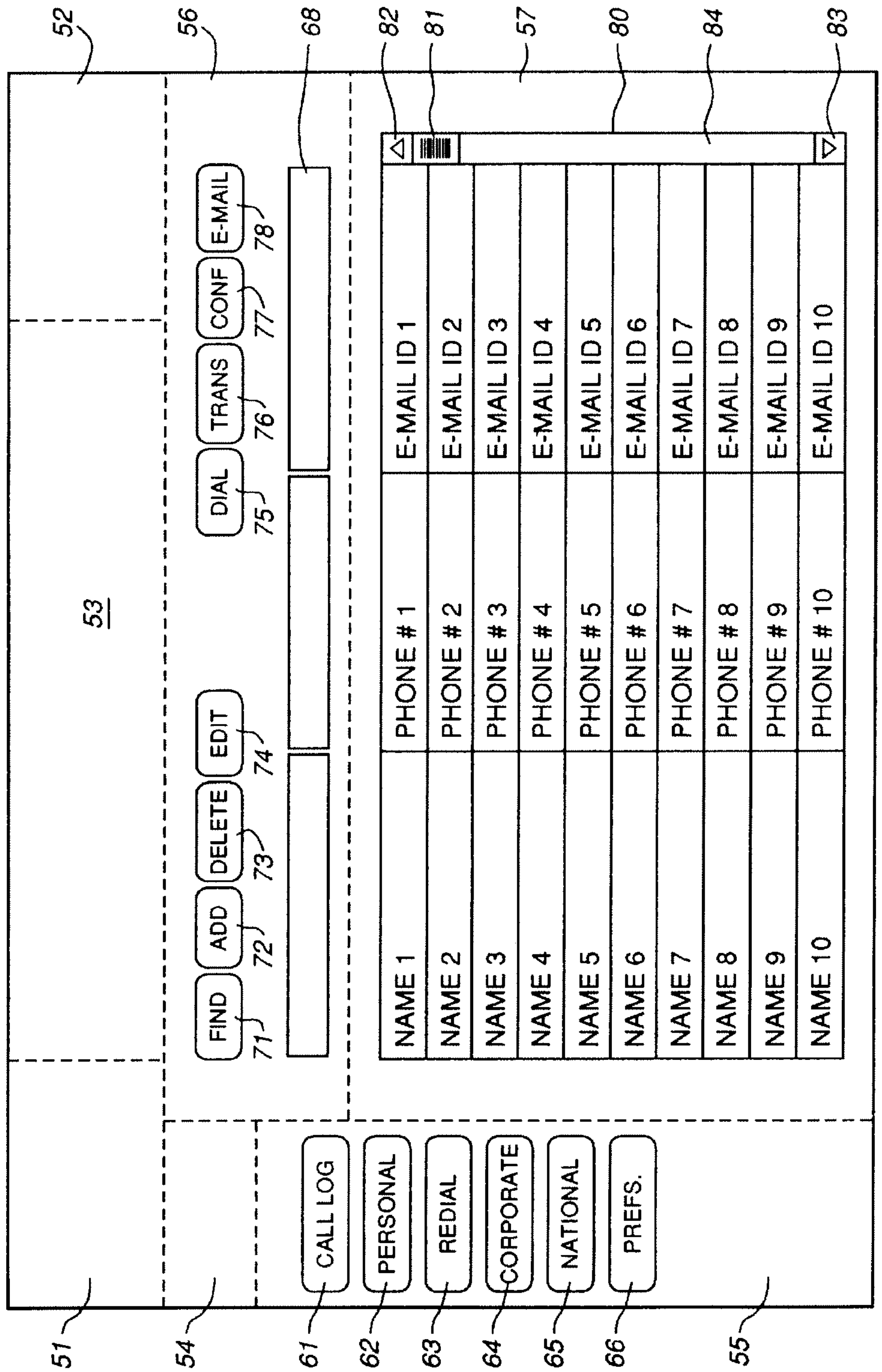

52
56
68
82
81
57
80
84
83
53
FIND
ADD
DELETE
EDIT
DIAL
TRANS
CONF
E-MAIL
71
72
73
74
75
76
77
78
NAME 1
NAME 2
NAME 3
NAME 4
NAME 5
NAME 6
NAME 7
NAME 8
NAME 9
NAME 10
PHONE # 1
PHONE # 2
PHONE # 3
PHONE # 4
PHONE # 5
PHONE # 6
PHONE # 7
PHONE # 8
PHONE # 9
PHONE # 10
E-MAIL ID 1
E-MAIL ID 2
E-MAIL ID 3
E-MAIL ID 4
E-MAIL ID 5
E-MAIL ID 6
E-MAIL ID 7
E-MAIL ID 8
E-MAIL ID 9
E-MAIL ID 10
CALL LOG
PERSONAL
REDIAL
CORPORATE
NATIONAL
PREFS.
51
54
61
62
63
64
65
66
55

Fig. 3

METHODS OF AND APPARATUS FOR PROVIDING TELEPHONE CALL CONTROL AND INFORMATION

This invention relates to methods of and apparatus for providing telephone call control and information.

BACKGROUND OF THE INVENTION

It is well known to provide relatively sophisticated telephone call control and information features using a subscriber's telephone. Some examples of telephone call control features are dialling of stored numbers, redialling of previously dialled numbers, three-way calling, and call forwarding. Examples of telephone information features are calling number display, calling number logs, and call waiting messages. Numerous other examples of call control and information features exist.

Providing such features using the subscriber's telephone involves several disadvantages. For example, the telephone must be capable of providing the required control input and information display functions, so that it becomes a relatively complicated and expensive device. As further call control and information features are developed and become available, the telephone may be unable to accommodate them so that it must be replaced or upgraded to permit use of these further features. Even when the necessary functions are present in the telephone, use of the various functions is not generally simple or intuitive, typically requiring the subscriber to enter various number sequences and/or to interpret relatively cryptic displayed messages. Furthermore, these functions are limited to each individual telephone device, and they must be provided separately for different telephone devices.

Some of these disadvantages have been avoided or reduced by the use of computer-telephone integration (CTI) software which is run on a subscriber's computer in association with telephone control hardware such as a modem or telephone dialler. Such software can facilitate the display of information to, and the input of control information by, the subscriber, and in addition to the features discussed above can facilitate the provision of other features such as telephone directories and voice massaging. However, these CTI arrangements also have several disadvantages. In particular, they require the use of a computer, software, and telephone control hardware by the subscriber, and the computer system must be running continuously to collect information on incoming calls. In addition, such arrangements only provide information at the location at which the system is installed.

More sophisticated arrangements are also known for use with private branch exchange (PBX) and key system telephone networks deployed over a local area network (LAN), with similar disadvantages.

It is also known to provide so-called web call center applications. In this case a subscriber uses a web browser, which for example may be constituted by software running on the subscriber's computer system, to access a computer network such as the international computer network generally referred to as the Internet or World Wide Web, which for brevity is referred to below simply as the web. On browsing a company's web site and desiring to talk with a customer representative of the company, the subscriber can enter his name and telephone number into an HTML (hypertext markup language) page and click on a "submit" button, in response to which the company's telephone system initiates a telephone call from an available representative back to the subscriber. Such call center applications do not provide the telephone call control and information features discussed above.

An object of this invention is to provide improved methods of and apparatus for providing telephone call control and information.

SUMMARY OF THE INVENTION

According to one aspect, this invention provides a method of making a telephone connection comprising the steps of: remotely accessing a computer network facility to produce at the computer network facility a telephone connection message including information identifying calling and called telephone numbers; communicating the telephone connection message from the computer network facility to a telephone switch via a switch-computer interface; and establishing a telephone connection between the calling and called telephone numbers from the switch in response to the telephone connection message.

The step of establishing a telephone connection preferably comprises the step of providing a ringing signal to a telephone identified by the calling telephone number.

Preferably the step of remotely accessing the computer network facility comprises providing telephone number information from the computer network facility for remote display to a subscriber identified by the calling telephone number. The telephone number information can comprise a personal telephone directory of the subscriber, and logged information relating to telephone communications to and/or from the calling telephone number. The step of remotely accessing the computer network facility conveniently comprises operating a web browser.

Thus the invention enables subscribers to control telephone connections, and obtain information from telephone directories and call logs, using a web browser without any need for extra hardware to couple the browser to the telephone. Call logs are maintained without requiring the browsers of the subscribers to be active. In addition, the web or computer network facility can be accessed by each subscriber from any location with web access facilities.

Another aspect of the invention provides a telephone call management system comprising: a computer network facility including a server for communications with telephone subscribers, an information database, and a telephone call control system; a telephone switch including a switch-computer interface; and a communications path between the telephone call control system of the computer network facility and the switch-computer interface of the telephone switch; wherein information relating to telephone communications to and/or from telephone numbers of subscribers is communicated via the communications path from the telephone switch to the computer network facility and being stored in the database, and information for controlling telephone communications is communicated via the communications path from the computer network facility to the telephone switch in response to remote access by subscribers to the information database via the server of the computer network facility.

The invention also provides a method of telephone call management, comprising the steps of: storing personal telephone directories and call logs of telephone subscribers for remote access by the subscribers via a web facility; supplying information, relating to at least some telephone communications associated with the telephone subscribers, from a telephone switch to the web facility; updating the personal telephone directories and call logs of the telephone subscribers in dependence upon information supplied by the subscribers by the remote access via the web facility and the information supplied from the telephone switch; and sup-

plying information from the web facility to the telephone switch, for controlling telephone communications for the subscribers via the telephone switch, in response to the remote access by the subscribers via the web facility.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will be further understood from the following description with reference to the accompanying drawings, in which:

FIG. **1** is a block diagram schematically illustrating an arrangement in accordance with an embodiment of the invention;

FIG. **2** is a block diagram schematically illustrating one form of a web facility of the arrangement of FIG. **1**; and

FIG. **3** illustrates an example of a web page layout which can be provided in the arrangement of FIGS. **1** and **2**.

DETAILED DESCRIPTION

Referring to FIG. **1**, in an arrangement in accordance with an embodiment of the invention a telephone subscriber has at least one telephone **10** and a web browser **12**. The telephone is coupled via a path **14** to a telephone switch **16**, and the web browser is coupled via a path **18** to a network **20** constituting the web (Internet or World Wide Web). The forms of the telephone **10**, web browser **12**, and paths **14** and **18** are entirely arbitrairy, and these can be known or yet to be devised. The telephone switch **16** can be a central office (C.O.) forming part of the public switched telephone network (PSTN), or a PBX or telephone key system which is coupled to the PSTN in a known manner.

For example, the telephone **10** can be a conventional telephone with pulse or DTMF (dual-tone multi-frequency) dialling, with or without any additional functions for control or information display, coupled to the telephone switch **16** via a twisted wire pair constituting the path **14**. For the purposes of this invention, it is observed that even the dialling function of the telephone **10** is not essential and can be dispensed with (although it would of course be required for conventional use of the telephone **10**). Alternatively, the path **14** could be provided via an ISDN (integrated services digital network) line or any other telephone communications path. With wireless communications, the telephone **10** can be a fixed or mobile telephone.

The path **18** can also be of any known or desired form, for example comprising a wireline or wireless data communications path which may be the same as or separate from the path **14**. Likewise, the form of the web browser **12** is entirely arbitrary. For example it may comprise a personal computer executing browser software in known manner, or a dedicated network browsing device, or a web browsing function integrated within another device such as a video game device or a television receiver or other communications device. Similarly, the functions of the web browser **12** and telephone **10** can be integrated into a single unit, with or without other functions, in any desired manner.

Thus there are numerous ways in which the telephone **10** and web browser **12**, and their paths **14** and **18**, can be implemented, for example including a conventional telephone and personal computer executing browsing software coupled via separate twisted wire pair telephone lines (or via a single telephone line using multiplexed communications) to the telephone switch **16** and web **20**, or an integrated mobile unit combining voice communication and network browsing functions coupled via wireless (e.g. infra-red or radio) communication paths to the web and the PSTN.

The arrangement of FIG. **1** also includes a web facility **22** that is coupled to, and thus can be considered as forming part of, the web **20**. Details of the web facility **22** are described below. The web facility **22** is also coupled via a path **24** to a switch-computer interface (SCI) **26** which forms part of the telephone switch **16**. The path **24** for example comprises a communications path providing X.25 communications between the web facility **22** and the SCI **26**, but it can alternatively comprise any other desired form of communications path, including for example an Ethernet communications path via the network or web **20**.

The SCI **26** is a known facility that is provided by the supplier of the telephone switch **16**. For example, in the event that the telephone switch is a DMS™ telephone switch available from Northern Telecom Limited, then the SCI **26** is constituted by CompuCALL™ facilities also available from Northern Telecom Limited for that switch. Other forms of SCI are available for other telephone switches. The interface **26** can use any of a variety of protocols, such as SCAI (Switch-Computer Access Interface), SPI (Service Programming Interface), or OAP (Open Automated Protocol). In addition, higher level interfaces, such as TSAPI (Telephony Server Application Programming Interface), TAPI (Telephony Application Programming Interface), or JTT (Java Telephony Toolkit) can be implemented in the SCI **26**, or in the CCI **46** described below. In any event, the SCI **26** provides on the path **24** information about telephone calls to telephone lines or directory numbers handled by the switch, and can also control the switch in response to control information supplied via the path **24** to establish calls as described further below.

The web facility **22** provides an interface to the subscriber, via the web **20**, path **18**, and browser **12**, in the form of one or more web pages that enable the subscriber to manage at least some and preferably all telephone functions for the telephone **10**. These functions for example can include all of the functions referred to in the introduction, some of which are further discussed below, as well as other functions which may be desired. To this end, the web facility **22** also communicates call control signals and information relating to these telephone functions with the telephone switch **16** via the path **24** and the SCI **26**.

Accordingly, the web facility **22** constitutes a web server interface for subscriber information and call management functions, and can have any form that enables these functions to be provided and that provides corresponding communications with the telephone switch **16** via the path **24** and SCI 26. FIG. **2** illustrates by way of example one form of the web facility **22**. This form of the web facility **22** comprises two computer systems, shown in FIG. **2** within dashed-line boxes and referred to below as a web system and a call control system **32**. By way of example, the web system **30** may comprise a Windows NT™ computer system and the call control system **32** may comprise a DEC Alpha 2100™ computer system.

The division of the web facility **22** between the two computer systems **30** and **32** is convenient for providing a security firewall between the public network **20** to which the web system **30** is connected and the private data within the computer system **32**, but all of the functions of the web facility **22** could alternatively be provided on a single computer system.

The web system **30** supports a web server **34**, a web page manager **36**, CGI (Communications Gateway Interface) scripts **38**, and a cache (working data storage) **40**. The web system **30** may also support an advertisement server (not

shown). The web server 34 and advertisement server are commercially available software applications which need not be described further here. The web page manager 36 is a software application that manages the presentation of the call management web pages to the subscriber via the web 20, and that can easily be provided in known manner to provide any desired web page appearance. Purely by way of example and illustration, FIG. 3 shows one possible appearance of a call management web page, and this is further described below. The CGI scripts 38 are software procedures that receive high-level calls from the web server 34 and translate these into lower level operations to be executed in conjunction with the cache 40 and the call control system 32, with parameters being passed to and from the CGI scripts accordingly.

The call control system 32 supports a database 42, call APIs (Application Program Interfaces) 44, and a call control interface 46. The call control interface 46 is a commercially available product, such as Genesys T-Server™, that provides a network or direct interface via the path 24 to the SCI 26 of the telephone switch 16. The call APIs 44 communicate with the CGI scripts 38 of the web system 30 via paths 48, and translate CGI script operations into low level operations comprising calls to and from the call control interface 46 and the database 42. Thus the CGI scripts 38 and call APIs 44 simply provide successively lower level procedures or software routines for handling calls between the web page manager 36 running on the web server 34, the call control interface 46, and the database 42 and cache 40. The database 42 comprises, for example, a commercially available database manager using SQL (structured query language) in a known manner.

The paths 48 are shown in FIG. 2 for convenience as direct paths between the CGI scripts 38 and the call APIs 44, but they are preferably constituted by InternetProtocol paths communicating remote procedure calls between these units.

Referring to FIG. 3, one possible appearance of a call management web page provided by the web facility 22 is illustrated. It is emphasized that this, and the following description of call management functions which can be provided, are given purely by way of example and explanation, and the invention is not in any way limited to these examples or the manner in which they are provided.

As shown by dashed lines in FIG. 3, the web page is divided into frames which are referenced 51 to 57. The frames 51 and 52 can be used to display logos relating to the call management service and its provider, and the frame 53 can be used to display an advertising banner. The advertising banner can be provided by an advertisement server on the web system 30 as indicated above or externally of the web facility 22 elsewhere on the web 20. The manner in which advertising banners are called, provided, and displayed is well known in the art and need not be described here.

The frame 54 can be used to display data relating to the subscriber, for example his name, telephone number, and e-mail ID (electronic mail identity), when he is logged on, and otherwise to display a message indicating that the subscriber is not logged on. The frame 55 provides a number of function buttons 61 to 66, each constituting a hypertext tag in known manner, functions of which can be as described below.

The frame 56 provides editing windows 68 and buttons 71 to 78, each constituting a hypertext tag, of which the buttons 71 to 74 provide directory functions and the buttons 75 to 78 provide communication functions as described below. The frame 57 provides contents dependent upon the functions selected by the buttons 61 to 66, as further described below.

On initially accessing the web facility 22, the web page manager 36 produces the web page for example with logos in the frames 51 and 52, an advertising banner obtained from the advertisement server in the frame 53, and with the frame 57 presenting options (e.g. function buttons and/or editing windows) to permit the subscriber to register or log in. On logging in, the web page manager 36 communicates via the functions 38 and 44 to retrieve data for the subscriber from the database 44 and store this data in the cache 40 for convenient and rapid access. This data can include subscriber information which the page manager 36 then displays in the frame 54 as indicated above, preferences previously stored for the subscriber, and personal directories and call data as discussed further below. The page manager 36 then can also present the frames 55 and 56 for example as shown in FIG. 3, with the frame 57 being blank or containing any desired information.

On clicking the button 62 labelled PERSONAL, via the function 38 the web page manager 36 accesses a personal directory of the subscriber and displays this in a conventional scrolling window 80 within the frame 57. For example as shown in FIG. 3 each entry in the personal directory can have name, telephone number, and e-mail ID fields which are displayed in the window 80. A slider 81, arrows 82 and 83, and scroll bar 84 permit the subscriber to scroll through the personal directory records. Clicking on any record causes the fields of that record to be reproduced in the editing windows 68, where the record can be edited and updated by clicking on the button 74 labelled EDIT. A record identified in the windows 68 can be deleted from the personal directory by clicking on the button labelled DELETE. A desired record can be located by the subscriber entering search criteria in the windows 68 and clicking on the button 71 labelled FIND, and new records can be created in the personal directory from the windows 68 by the subscriber clicking on the button 72 labelled ADD. In this manner, via the web page manager 36 and the CGI scripts 38, the subscriber can set up and maintain the personal directory in the cache 40. Updating of the database 42 from the cache 40 can be carried out as desired in the background in a known manner.

The above functions of the buttons 71 to 74 do not involve communications via the call control interface 46. In contrast, the buttons 75 to 77 invoke communications functions which typically involve communications with the telephone switch 16 via the call control interface 46. For example, clicking on the button 75 labelled DIAL triggers the telephone switch 16 to set up a telephone connection between the subscriber's telephone 10 and a telephone directory number in the windows 68. This number can be entered and optionally edited by the subscriber by typing at the network browser 12, selected from the personal directory by clicking on a record in the window 80 as described above, or provided in another manner for example as described further below.

On clicking the DIAL button 75, the web page manager 36 communicates a message, containing a dial request, a calling telephone number CN of the subscriber (as displayed in the frame 54), and a called telephone number DN from the windows 68, via the functions 38 and 44 to the call control interface 46, via which this message is forwarded via the path 44 and SCI 46 to the telephone switch 16. The switch 16 checks validity of the telephone numbers and that the subscriber's telephone 10 (calling telephone number CN) is on-hook, and provides a (possibly distinctive) ringing signal to the telephone 10. The subscriber, expecting this ring signal, takes his telephone 10 off-hook, and this is detected

by the telephone switch **16** in conventional manner, in response to which the switch **16** sets up the desired telephone connection to the called number DN in the same manner as if the number DN had been dialled by the subscriber at the telephone **10**. Error and/or status messages can be communicated from the telephone switch **16** via the SCI **46**, path **44**, and functions **46**, **44**, and **38** to the web page manager **36**, and displayed on the web page, as desired and appropriate.

It can be appreciated that, in the manner described above, the subscriber is able to instigate a telephone call to a desired number through his access to the web page, and not by dialling at the telephone **10**.

In a corresponding manner, the subscriber can transfer an existing telephone call at his telephone **10** to another called number DN in the windows **68** by clicking the button **76** labelled TRANS. The web page manager again communicates the numbers CN and DN, with a call transfer request, to the telephone switch **16** via the functions **38**, **44**, **46**, and **26**, in response to which the switch **16** transfers the call from the telephone **10** (CN) to the called number (DN) and provides error and/or status messages to the web page manager **36** accordingly. Likewise, the subscriber can establish a conference connection to add another called number DN from the windows **68** to an existing telephone call at his telephone **10** by clicking the button **77** labelled CONF. The web page manager again communicates the numbers CN and DN, with a conference request, to the telephone switch **16** via the functions **38**, **44**, **46**, and **26**, in response to which the switch **16** establishes a conference connection of the call involving the telephone **10** (CN) with the additional called number (DN, again providing en-or and/or status messages to the web page manager **36** accordingly.

In response to clicking on the button **78** labelled E-MAIL, the web page manager **36** creates in known manner a window for the subscriber to enter an e-mail message to an e-mail ID from the window **68** or entered by the subscriber in the e-mail window, this being transmitted in known manner via the web **20**. In this manner, electronic mail communications can also be established by the subscriber using the same web interface as for telephone voice communications. Other communications facilities, for example voice mail messages, and other telephony functions, can be similarly provided in analogous manner to the specific examples given above.

For telephone calls incoming to the telephone **10** via the telephone switch **16**, the SCI **26** provides to the web facility **22** information messages containing for example the called and calling numbers, and the date and time of the call. This information is entered by the call APIs **44** into a call log for the respective subscriber in the database **42** via the functions **46** and **44**. This takes place whether or not the subscriber's web browser **12** is active, so that the call log is not dependent on any activity of the subscriber. On log-in to the web page, the call log is supplied to the cache **40** as described above and is available to the subscriber. The subscriber can click the button **61** labelled CALL LOG, in response to which the web page manager **36** displays the call log in a scrolling window in the frame **57** in place of the personal directory. Each record in the call log can for example include a field containing the calling telephone number (e.g. as in the personal directory described above) supplied from the telephone switch **16** via the SCI **26**, a field for a name which can be optionally provided either similarly by the telephone switch **16** by look-up from the calling telephone number, using the subscriber's (i.e. the called number's) personal directory via the database **42** or using other directory facilities such as a corporate directory as discussed below, and a field for the date and time of the call. Other fields, for example for the duration and status (e.g. answered or not) of the call provided by the SCI **26**, and an associated e-mail address as described above and also provided by the database or directory lookup, can also be provided in the call log as desired.

In a similar manner to that described above for the personal directory, the subscriber can scroll through the call log, click on any record to reproduce it in the windows **68**, click on the ADD button **72** to add a corresponding record to the subscriber's personal directory, click on the DIAL button **75** to establish a return call to the calling number, etc.

For alerting the subscriber to an incoming telephone call, a ringing signal is supplied to the telephone **10** in conventional manner. In addition, if the subscriber's web browser (or a sub-set of this such as a Java applet) is active, then the web page manager **36** is supplied with information about the call (e.g. calling number, name, etc. as provided for the call log as described above) via the functions **44** and **38** and provides an informative alert to the subscriber's web page (or applet window). This obviates the need for processing and display facilities in the telephone **10** to provide call information.

Correspondingly, the web facility maintains a called number log, of numbers called by the subscriber. Conveniently this can be similar to the call log described above, and for calls established by the subscriber using the web facility can use information supplied from the web page manager **36** and/or information supplied by the switch **16** via the SCI **26** as described above for incoming telephone calls to the subscriber. The latter information can also be used to maintain this called number log even for calls made in conventional manner from the telephone **10** without use of the web facility **22**, so that the subscriber's web browser **12** does not need to be active for this called log to be maintained. In the same manner as described above for the call log, the subscriber can click on the button **63** labelled REDIAL to display the called number log in a window in the frame **57**, and again the windows **68** and buttons **71** to **78** in the frame **56** can be used by the subscriber to maintain the called number log and, using the DIAL button **75**, to redial previously called numbers.

A directory of employees of a corporation can be maintained by the web facility **22**, for example as part of the database **42**, and can be used as described above to determine names and other information corresponding to supplied telephone numbers. In addition, such a directory can be used generally by the subscriber using the web facility **22**. In response to the subscriber clicking the button **64** labelled CORPORATE, the web page manager **36** in this case presents in the frame **57** a corporate directory search window in which the subscriber can enter search criteria to locate information for anyone in the directory. Such information is then displayed by the web page manager **36** in the frame **57**, for example in a similar manner to the display of the subscriber's personal directory in this frame as described above and illustrated in FIG. **3**. As in that situation and the other situations described above, the subscriber can scroll through the directory information, click on any record to reproduce it in the windows **68**, click on the ADD button **72** to add a corresponding record to the subscriber's personal directory, click on the DIAL button **75** to establish a call to the selected number, etc.

Other directories can also be accessed by the subscriber via the web facility **22**. For example, a national telephone

directory, containing names, addresses, and telephone numbers, maintained elsewhere on the web **20** can be used by the subscriber by clicking on the button **65** labelled NATIONAL. In response to this the web page manager establishes an http (hypertext transfer protocol) link via the web **20** to the web site of the national directory in a known manner, and presents its search window to the subscriber in the frame **57**. As in the case described above, the subscriber can then find information using the national directory, copy and paste it or click on it to reproduce it in the windows **68**, and add the information to the subscriber's personal directory by clicking on the ADD button **72**, dial the number by clicking on the DIAL button **75**, etc. Other directories external to the web facility **22** can be similarly accessed.

The subscriber can also click on the button **66** labelled PREFS., in response to which the web page manager **36** presents in the frame **57** options for the subscriber to set preferences for his use of the web facility **22** in a known manner.

It will be apparent to those of ordinary skill in the art that all of the above functions, and many other functions which may be desired, can be provided in a relatively straightforward manner by simple messages or procedure calls and responses, with appropriate parameters and returned values, between functions of the web facility **22**, and specifically between the web page manager **36** and the call control interface **46**, cache **40**, and database **42** via the CGI scripts **38** and the call APIs **44**. Details of these procedures, parameters, and returned values depend on the particular functions that are provided and the particular manner in which the web facility **22** is implemented. Such details can be routinely determined by persons of ordinary skill and knowledge, and accordingly need not be, and are not, described here.

A number of significant advantages that may not be immediately apparent can be provided by embodiments of the invention. It can be appreciated that a subscriber can use any web browser **12** and any telephone **10** to provide all of the functions which are available via the web facility **22**. Neither of these is required to have any special hardware or software features, beyond very basic capabilities of the telephone **10** and the inherent functioning of the web browser **12**. The subscriber does not need to acquire or maintain any other software or hardware.

The subscriber is able to access his telephone web page on the web facility **22** from any web browser at any location. This enables all of his call management functions to be available to him regardless of where he may be, for example at home, in an office, or travelling using a mobile telephone and web browser. A particular advantage of this is provided if one of the telephone functions available to the subscriber is call forwarding. In this case for example the subscriber can access the web facility **22** from his office, activate via the web facility **22** a call fowarding function which causes the telephone switch **16** to redirect to his office telephone number calls that are directed to his home telephone number, and receive such calls at his office. Conversely, on returning home he can again access the web facility **22** to remove the call forwarding. The web facility **22** controls the telephone switch to effect and remove the call forwarding function in a similar manner to that described above for call transfer, using another button and related procedures to perform these functions.

Numerous other communications functions can be similarly and easily provided in a corresponding manner, and as already stated, the above description is given purely by way of illustration of the functions that may be provided. As can be appreciated, further functions (both known and yet to be developed) can be easily added by the web facility **22**, and these can be made available immediately to the subscriber, possibly on a subscription or pay-per-use basis that enhances revenues to the service provider. Obviously, the same web facility can be used to serve an arbitrary number of subscribers.

Thus although a particular form of the invention has been described above, it can be appreciated that numerous modifications, variations, and adaptations may be made without departing from the scope of the invention as defined in the claims.

What is claimed is:

1. A method of making a telephone connection comprising the steps of:

storing, for access by a computer network facility which is remotely accessible using a web browser, telephone number information relating to a telephone subscriber;

remotely accessing the computer network facility using a web browser for display of said telephone number information to the subscriber;

producing at the computer network facility using the web browser a telephone connection message including information identifying a calling telephone number of the subscriber and a called telephone number;

communicating the telephone connection message from the computer network facility to a telephone switch via a switch-computer interface; and

establishing a telephone connection between the calling and called telephone numbers from the switch in response to the telephone connection message.

2. A method as claimed in claim **1** wherein the step of establishing a telephone connection comprises the step of providing a ringing signal to a telephone identified by the calling telephone number.

3. A method as claimed in claim **1** wherein the telephone number information comprises a personal telephone directory of the subscriber.

4. A method as claimed in claim **1** wherein the telephone number information comprises logged information relating to telephone communications to and/or from the calling telephone number.

5. A method as claimed in claim **1** and further comprising the step of:

communicating information relating to telephone communications to and/or from the calling telephone number from the switch to the computer network facility;

wherein said telephone number information includes said information communicated from the switch.

6. A method as claimed in claim **5** wherein the telephone number information includes personal telephone directory information of the subscriber.

7. A method as claimed in claim **6** wherein the step of establishing a telephone connection comprises the step of providing a ringing signal to a telephone identified by the calling telephone number.

8. A telephone call management system comprising:

a computer network facility including a web server for communications with web browsers of telephone subscribers, an information database for storing telephone number information relating to the subscribers, and a telephone call control system;

a telephone switch including a switch-computer interface; and

a communications path between the telephone call control system of the computer network facility and the switch-computer interface of the telephone switch;

wherein information relating to telephone communications to and/or from telephone numbers of subscribers is communicated via the communications path from the telephone switch to the computer network facility and is stored in the database for the respective subscribers, and information for controlling telephone communications is communicated via the communications path from the computer network facility to the telephone switch in response to remote access by the respective subscribers to the information database via web browsers of the respective subscribers and the web server of the computer network facility.

9. A system as claimed in claim **8** wherein information stored in the database comprises telephone numbers calling and/or called by the telephone subscribers.

10. A system as claimed in claim **9** wherein information stored in the database further comprises personal telephone directories of the telephone subscribers.

11. A system as claimed in claim **8** the information for controlling telephone communications communicated from the computer network facility to the telephone switch comprises information identifying a telephone number of a subscriber remotely accessing the server of the computer network facility and information identifying a telephone connection request and another telephone number associated with the request.

12. A method of telephone call management, comprising the steps of:

storing personal telephone directories and call logs of telephone subscribers for remote access by the subscribers via a web facility;

supplying information, relating to at least some telephone communications associated with the telephone subscribers, from a telephone switch to the web facility;

updating the personal telephone directories and call logs of the telephone subscribers in dependence upon information supplied by the subscribers by the remote access via the web facility and the information supplied from the telephone switch; and

supplying information from the web facility to the telephone switch, for controlling telephone communications for the subscribers via the telephone switch, in response to the remote access by the subscribers via the web facility.

13. A method as claimed in claim **12** wherein the information supplied from the telephone switch to the web facility identifies calling and called telephone numbers of the telephone subscribers.

14. A method as claimed in claim **12** wherein the information supplied from the web facility to the telephone switch identifies subscriber telephone numbers and connection requests identified by the subscribers by the remote access via the web facility.

* * * * *

EXHIBIT B

US006445695B1

(12) **United States Patent**
Christie, IV

(10) **Patent No.: US 6,445,695 B1**
(45) **Date of Patent: Sep. 3, 2002**

(54) **SYSTEM AND METHOD FOR SUPPORTING COMMUNICATIONS SERVICES ON BEHALF OF A COMMUNICATIONS DEVICE WHICH CANNOT PROVIDE THOSE SERVICES ITSELF**

(75) Inventor: **Samuel H. Christie, IV**, Cary, NC (US)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/223,880**

(22) Filed: **Dec. 31, 1998**

(51) **Int. Cl.**[7] **H04L 12/66**; H04J 3/16

(52) **U.S. Cl.** **370/352**; 370/466

(58) **Field of Search** 370/264, 351–4, 370/389, 395.1, 395.61, 395.5, 419, 463, 465–469, 400–2

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,031,896 A * 2/2000 Gardell et al. 370/352
6,161,008 A * 12/2000 Lee et al. 370/352
6,229,804 B1 * 5/2001 Mortsolf et al. 370/352
6,259,691 B1 * 7/2001 Naudus 370/352
6,275,574 B1 * 8/2001 Oran 370/352
2001/0026548 A1 * 10/2001 Strathmeyer et al. 370/356
2001/0028649 A1 * 10/2001 Pogossiants et al. 370/389
2001/0028654 A1 * 10/2001 Anjum et al. 370/401
2002/0018464 A1 * 2/2002 Kikinis 370/352

* cited by examiner

Primary Examiner—David Vincent
(74) *Attorney, Agent, or Firm*—Withrow & Terranova, P.L.L.C.

(57) **ABSTRACT**

A system and method for supporting communications services on behalf of a communications device which cannot provide those services itself in a communications network based on functional signaling. A terminal is designed to identify a supporting server/terminal proxy upon initialization. Henceforth, the terminal provides each user input stimulus to the server and responds to stimulus from the server. The server manages the state machine of the terminal, provides supplementary services, and meets protocol requirements for the network interface.

16 Claims, 2 Drawing Sheets

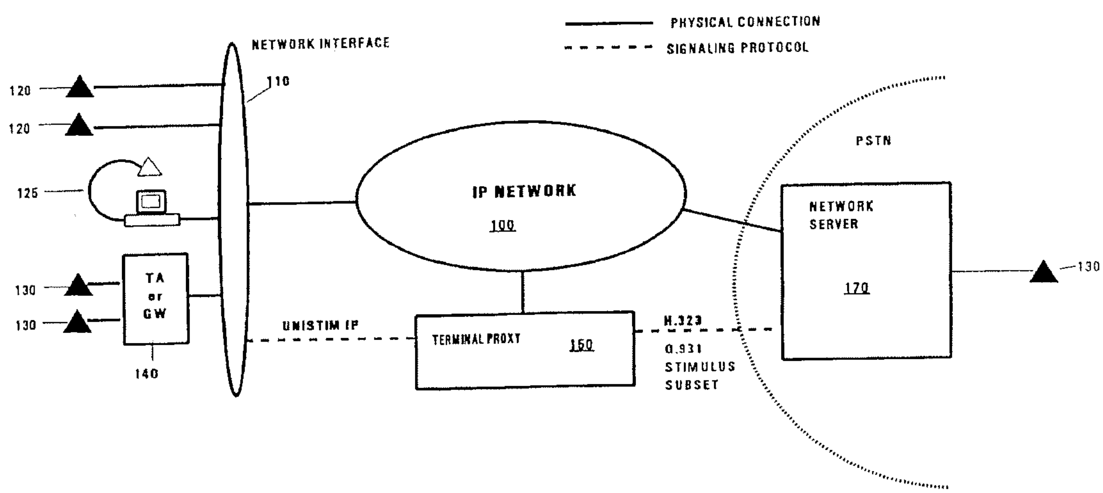


FIG. 1

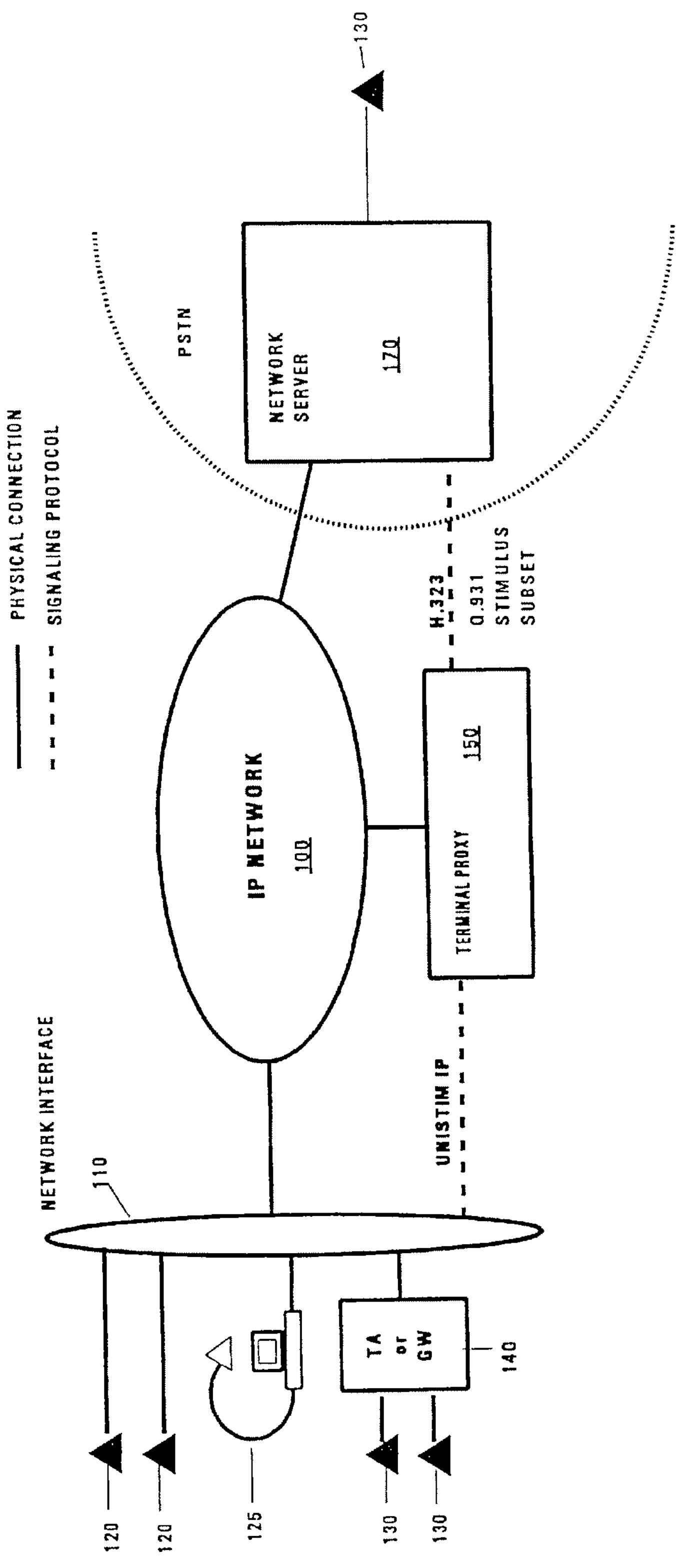
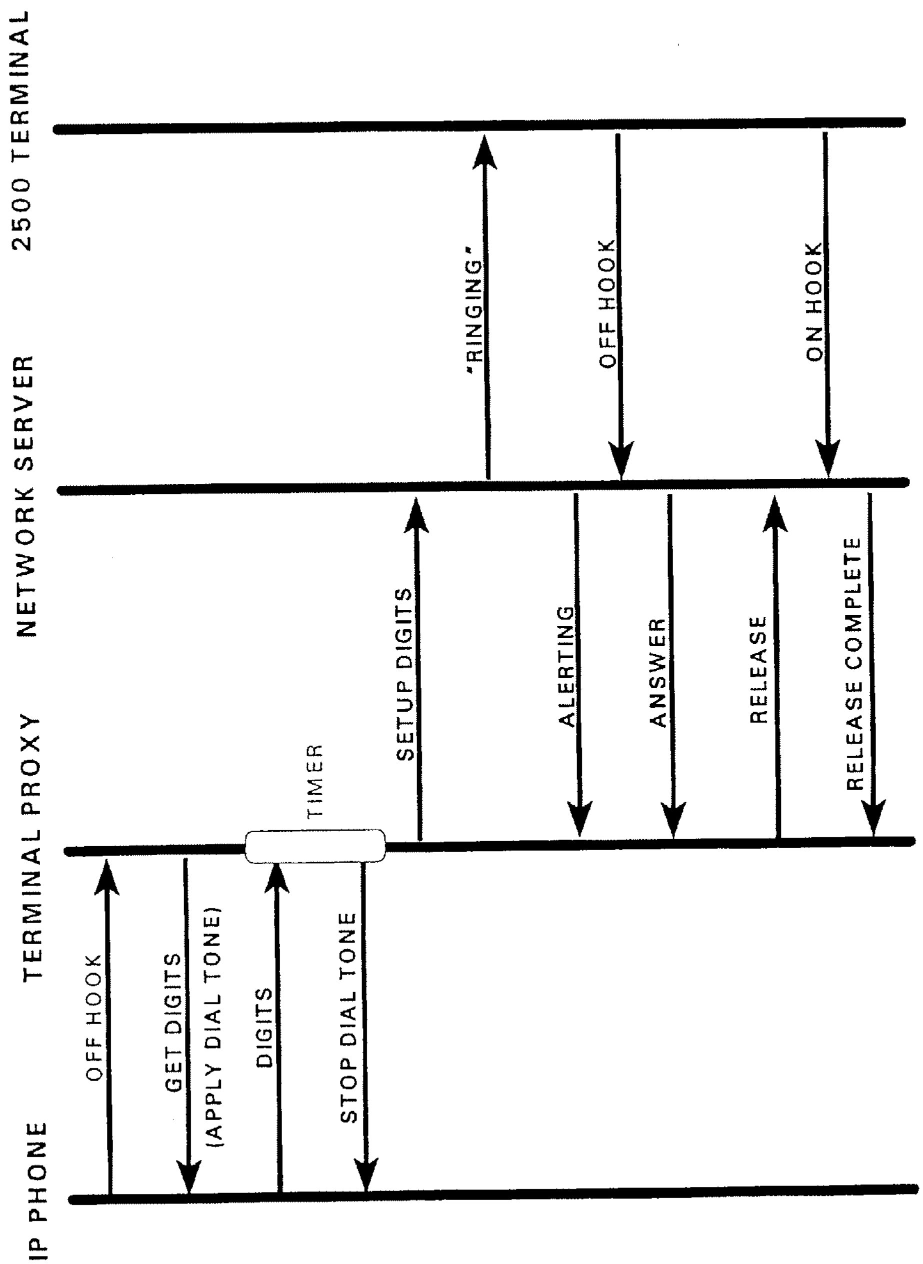

130
PSTN
NETWORK
SERVER
170
PHYSICAL CONNECTION
SIGNALING PROTOCOL
H.323
Q.931
STIMULUS
SUBSET
IP NETWORK
100
TERMINAL PROXY
150
NETWORK INTERFACE
110
UNISTIM IP
TA
or
GW
140
120
120
125
130
130

FIG. 2


IP PHONE
TERMINAL PROXY
NETWORK SERVER
2500 TERMINAL
OFF HOOK
GET DIGITS
(APPLY DIAL TONE)
DIGITS
STOP DIAL TONE
TIMER
SETUP DIGITS
"RINGING"
ALERTING
OFF HOOK
ANSWER
RELEASE
ON HOOK
RELEASE COMPLETE

SYSTEM AND METHOD FOR SUPPORTING COMMUNICATIONS SERVICES ON BEHALF OF A COMMUNICATIONS DEVICE WHICH CANNOT PROVIDE THOSE SERVICES ITSELF

TECHNICAL FIELD

The present invention relates to a system and method for supporting communications services on behalf of a communications device which cannot provide those services itself. More specifically, the present invention relates to a system and method for supporting foreign terminals in a communications network.

BACKGROUND AND RELATED ART

In the developing field of IP telephony, there are a multitude of network architecture designs and implementations to consider. The ultimate goal, from a subscriber standpoint, is to provide an IP telephony system that does not require the subscriber to alter the way they perform existing tasks such as placing and receiving calls, and does not affect the set of calling services (caller ID, call forwarding, etc.) that the subscriber may have subscribed to. That is, any new IP telephony system should operate such that subscribers do not have to learn new means for accessing and using services they have always used from their traditional phones should those phones be so called "dumb" terminals.

IP telephony protocols, and there are several under industry-wide consideration, and existing telephony protocols are quite different due to the fact that each handles voice and/or data signals differently. The chief difference is that an IP network partitions a voice and/or data signal into packets and relays the packets over the network from point to point. The packets are then re-arranged at the endpoint for distribution to the consumer in a usable form. Most existing telephony networks are analog in nature meaning signals are not broken into digital packets.

Moreover, IP telephony networks and existing telephony networks must be made compatible with one another in

order to allow an IP telephony subscriber to communicate with a non-IP telephony subscriber and vice-versa. This necessitates network interfaces capable of converting between IP standards and protocols and existing standards and protocols.

Existing telephony networks have an advantage over IP telephony networks in that extensive call processing network hardware is already in place. Moreover, calling services have been developed to operate within existing telephone architecture. Thus, it makes sense to develop an IP telephony system that can utilize to the greatest extent possible the existing network architecture. This primarily entails upgrading certain existing hardware with the previously mentioned network interfaces capable of converting between IP standards and protocols and existing standards and protocols. Another option is to create standalone network interface devices that perform protocol conversion. Such devices would then augment existing architecture. It is to be understood that some of the protocol conversion between IP telephony and existing telephony can be achieved via software. That is, a 100% hardware solution is not required.

As mentioned before, an IP telephony subscriber may want to access the same calling services as existing telephony subscribers. There are a multitude of such services that are currently offered by telephony service providers. These services all require some form of call processing logic to achieve their stated goal. Some services require more extensive processing than others and therefore are network based meaning that the processing logic is performed in a device or node within the telephone network and not by the phone or terminal itself. Other less computationally intensive services may be by the consumer's own terminal (phone). Additionally, some terminals are "smarter" than others in that they possess greater processing ability and can, therefore, perform certain services themselves as opposed to relying on the network. The table below provides a sampling of calling services and the location at which they may be performed, that is whether the service is network or terminal based. The following table is exemplary, not exhaustive.

TABLE 1

CALLING SERVICE	LOCATION WHERE PERFORMED
CALL FORWARDING NO RESPONSE	NETWORK BASED
CALLING NUMBER DELIVERY/BLOCKING	NETWORK BASED
CLOSED USER GROUP	NETWORK BASED
LOCAL NUMBER PORTABILITY	NETWORK BASED
CONNECTED NUMBER DELIVERY/BLOCKING	NETWORK BASED
EMERGENCY CALL	NETWORK BASED
MALICIOUS CALL TRACE	NETWORK BASED
CALL MONITORING	NETWORK BASED
RELEASE LINE TRUNK	NETWORK BASED OR TERMINAL BASED
ACCOUNT CODE	NETWORK BASED OR TERMINAL BASED
EXTENSION SERVICES	NETWORK BASED OR TERMINAL BASED
MULTIPARTY	NETWORK BASED OR TERMINAL BASED
CALLING NAME DELIVERY	NETWORK BASED OR TERMINAL BASED
800 QUERY	NETWORK BASED OR TERMINAL BASED
IN SERVICE SWITCHING AND RESOURCE FUNCTIONS	NETWORK BASED OR TERMINAL BASED
HOLD AND RETRIEVE	TERMINAL BASED
CALL SCREENING	TERMINAL BASED
CALL TRANSFER	TERMINAL BASED
CALL FORWARDING	TERMINAL BASED
CALL WAITING	TERMINAL BASED
CALL RETURN	TERMINAL BASED
SPEED DIALING	TERMINAL BASED
REPEAT DIALING	TERMINAL BASED

New IP telephony networks define interfaces between a terminal (or client) and a network which necessitate a computationally powerful terminal. The terminal must be capable of managing its call state, providing supplementary services, and managing bearer connections, etc. All of these responsibilities require computational processing capacity (hardware) and software logic. Large numbers of terminals will likely be installed into a network which is expensive in terms of hardware. It is possible to create less computationally intense, less expensive terminals. However, the responsibilities of these scaled back terminals still must be supported. The present invention provides for the delegation of certain terminal responsibilities to a server residing on a network. Thus, use can be made of less expensive, less computationally intensive terminals in an IP telephony network.

The present invention is to be distinguished from IP PBX systems and "black box" devices which allow analog phones to be connected to digital PBX systems. The key difference between a terminal proxy as disclosed by the present invention and an IP PBX controller is that the terminal proxy is not a network call processing engine, rather it is a remote implementation of local call processing. The terminal proxy makes a terminal look like a terminal of another type from the perspective of the IP PBX controller or the central office. The key differentiation with respect to the black boxes which permit analog phones to be connected to digital PBXs is that the black box processes the media and is physically between the PBX and the phone. The terminal proxy, in contrast, is a signaling translator which is not physically between the two machines, only logically so.

SUMMARY OF THE INVENTION

The present invention applies traditional telephony architecture and design to an IP telephony network. To date, thin client (simple terminal) solutions have not been proposed for IP telephony, except in such a way as would eliminate service capabilities or bundle them in a network server such as, for instance, an IP PBX controller.

A terminal according to the present invention is designed to identify its supporting server upon initialization. Henceforth, the terminal provides each user input stimulus to that server (or its backup) and responds to stimulus from the server via a network interface to an IP network using, for instance, the NORTEL proprietary universal stimulus IP protocol (hereinafter "UNISTIM IP"). The server is then responsible for managing the state machine of the client, providing local or terminal specific supplementary services, and meeting protocol requirements for, inter afia, the network interface, SIP, or H.323.

The present invention thus comprises at least one terminal coupled to an IP network and a terminal proxy coupled to the IP network and communicable with the terminals. The terminal proxy communicates with and manages the call processing logic for terminals which cannot perform such tasks for themselves. The present invention further comprises a network interface situated between the terminals and the IP network interface for ensuring that all call control functional signaling between the IP network and the terminals are in a compatible format. The present invention still further comprises a terminal adapter coupled to the IP network via the network interface and supports terminals that do not communicate in an IP protocol. The terminal adapter receives the call control protocol of the terminal and converts it into an IP protocol usable by the IP network.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic illustrating one possible network architecture for the present invention; and

FIG. 2 is a message sequence diagram illustrating the call control processing utilized by the terminal proxy methodology of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

The present invention provides a logical means for remote implementation of local call processing that supports calling services and manages call state information and bearer connections in an IP telephony environment on behalf of a terminal which cannot provide those services itself. The premise involves the use of a terminal proxy hosting a client call server residing on the IP network and communicable with "dumb" terminals attached to the IP network. Dumb terminals may take the form of CLASS phones, 2500 sets, IP enabled phones without a high level of computational power, or PCs running IP telephony software. If the terminal happens to be a 2500 set or CLASS phone, then it must be connected to a terminal adapter or gateway mechanism in order to present the proper interface to the IP network. In any event, these dumb terminals will rely on the terminal proxy to perform signaling translation for certain call processing functions. As such, the terminal proxy is not physically between a terminal and an IP PBX. Rather it is logically between the two machines. That is, the terminal proxy makes a terminal appear as another terminal from the perspective of an IP PBX controller.

FIG. 1 illustrates an example of a system architecture for use in the present invention. An IP network **100** is connected to a number of terminal devices through a network interface **110**. The terminal devices illustrated include IP enabled phones **120**, compound terminal(s) **125** comprising more than one physical device such as, for instance, a phone and a computer, and standard 2500 terminals **130**. The 2500 terminals **130** are coupled to the network interface **110** via a terminal adapter/gateway **140**. IP network **100** is also coupled to a terminal proxy **150**. The terminals **120, 125, 130** become logical entities to the remainder of the telephony network via terminal proxy **150**.

Additionally, IP network **100** is connected to a network telephony server **170**. Terminals **120, 125, 130** communicate with terminal proxy **150** via network interface **110** using a stimulus protocol, most likely UNISTIM IP. UNISTIM IP assumes a call control architecture in which the call control "intelligence" lies outside of the telephone and is thus handled by external call control elements. In the present invention, the call control intelligence is handled either in the terminal itself (rudimentary functions), the terminal proxy **150**, or in network server **170** (complex functionality). An implementation of network server **170** may include a gateway and a PSTN central office switch, PBX, H.323 gatekeeper, or SIP proxy.

Terminal proxy **150** and network server **170** communicate over IP network **100** using the H.323 protocol. The primary H.323 telephony network lelements include a gatekeeper, a terminal, and a gateway. A gatekeeper (GK) is the network entity responsible for IP network address resolution and bandwidth allocation. Terminals provide for real-time two-way communication with one another. The gateway (GW) is the network entity that provides an interface to a non-H.323 network such as, for instance, a public switching telephone network (PSTN). These elements when combined form an H.323 Zone. There may be several gateways in an H.323 zone, each of which provide an interface into some non-H.323 network. The terminals and the gateways are the endpoints of the zone and the gatekeeper manages communications between endpoints of the zone.

To support a plain old telephone service (POTS) terminal **130**, e.g., a 2500 set, in the H.323 IP network **100**, additional network elements are required. POTS phones are connected to IP network **100** via a terminal adapter (TA) **140**. Signaling between POTS terminals and the terminal adapter complies with LATA Switching Systems Generic Requirements (LSSGR). Terminal adapter **140** interacts with the terminal proxy (TP) **150** based on a Stimulus Protocol, such as, for instance SGCP. The stimulus protocol is a messaging system that describes POTS terminal operations. Terminal proxy **150** logically supplies the functional intelligence of an H.323 compliant terminal on behalf of terminal adapter **140** and its supported terminals **130**.

The terminal proxy **150** is a software entity that provides the intelligence and processing for calling services, including basic call processing, on behalf of less capable terminals such as the IP phones **120** and 2500 sets **130** illustrated in FIG. **1**. While the terminal proxy logic could be located on the customer premise, potentially running on the terminal adapter processor, a centralized implementation reduces cost, increases reliability, and facilitates administration. Thus, terminal proxy **150** is a logical remote implementation of local call processing.

One embodiment uses UNISTIM IP as the signaling protocol between terminal adapter **140** and terminal proxy **150** while a stimulus subset of the Q.931 protocol is for communication between terminal proxy **150** and network server **170**.

FIG. **2** illustrates the message sequencing that occurs when an IP phone user initiates a call with a 2500 terminal. An IP Phone subscriber lifts his handset thereby initiating an off hook message which is sent to the terminal proxy. The terminal proxy then sends a return message requesting application of a dial tone and awaits digit entry from the subscriber. The terminal proxy also starts a timer with sufficient time allotted for the subscriber to enter the digits. The subscriber then inputs the digits of the directory number of the terminal that he wishes to call. Upon receipt of the correct number of digits, the terminal proxy sends a message back to the IP phone to stop the dial tone followed by a setup message to the network server serving the IP phone. The setup message informs the network server that a call to the end terminal 2500 set is being made. The network server responds by ringing the desired 2500 literminal while also sending an alerting message back to the terminal proxy. The terminal proxy sends a message requesting a ringback be applied to the IP phone to inform the calling party that the 2500 terminal is ringing. When the called party answers the 2500 terminal, an off hook message is sent to the network server. The network server then forwards an answer message to the terminal proxy. The terminal proxy sends a message request to stop the ringback and open a media channel for cut through audio. The two parties can now engage in a conversation. When the conversation is complete and the IP phone is placed back in its cradle, an on hook message is sent to the terminal proxy signifying the call is over. The terminal proxy responds by sending a release message to the network server. Once the 2500 terminal has "hung up" an on hook message is sent to its network server. The network server responds by sending a release complete message back to the terminal proxy indicating that the calling and called parties have concluded their call and their connection has been terminated. Both terminals are now placed back into a ready state.

The foregoing example illustrates the concept of having the terminal proxy manage the call on behalf of an IP phone. Similar processing would occur had the calling party used a 2500 terminal, or the like, connected to a terminal adapter. In such a case, the terminal adapter is required to interpret the 2500 terminal's POTS operations.

Call state information, bearer connections, and calling services are handled in the terminal proxy residing on the IP network rather than at the terminal (IP phone). To implement such a system it is necessary for the terminal to identify its supporting server within the IP network upon initialization. This is achieved utilizing standard operations that are well known in the art such as, for instance, DHCP or direct keying into the terminal.

It is to be understood that each of the method or process steps illustrated herein are readily implementable by those of ordinary skill in the art as a computer program product having a medium with a computer program embodied thereon. The computer program product is capable of being loaded and executed on the appropriate computer processing device(s) in order to carry out the method or process steps described. In addition, appropriate computer program code in combination with hardware implements many of the elements of the present invention. This computer code is often stored on storage media. This media can be a diskette, hard disk, CD-ROM, or tape. The media can also be a memory storage device or collection of memory storage devices such as read-only memory (ROM) or random access memory (RAM). Additionally, the computer program code can be transferred to the appropriate hardware over some type of data network.

In the claims, means-plus-function clause are intended to cover the structures described herein as performing the recited function and not only structural equivalents but also equivalent structures. Therefore, it is to be understood that the foregoing is illustrative of the present invention and is not to be construed as limited to the specific embodiments disclosed, and that modifications to the disclosed embodiments, as well as other embodiments, are intended to be included within the scope of the appended claims. The invention is defined by the following claims, with equivalents of the claims to be included therein.

It is to be further understood that the foregoing is illustrative of the present invention and is not to be construed as limited to the specific embodiments disclosed, and that modifications to the disclosed embodiments, as well as other embodiments, are intended to be included within the scope of the appended claims. The invention is defined by the following claims, with equivalents of the claims to be included therein.

What is claimed is:

1. A client proxy signaling system for an IP telephony network comprising:

at least one terminal coupled to an IP network; and

a terminal proxy coupled to said IP network communicable with said at least one terminal for providing logical call processing signaling on behalf of said at least one terminal,

wherein said terminal proxy communicates with and manages the call processing logic for said at least one terminal with respect to the remainder of a telephony network.

2. The system of claim **1** further comprising a terminal adapter coupled to said IP network and said terminal proxy and supporting at least one terminal that does not communicate in an IP protocol, said terminal adapter for receiving the call control protocol of the at least one terminal and converting same into an IP protocol usable by the IP network.

3. The system of claim **2** wherein said IP protocol between the terminal adapter and the terminal proxy is the simple gateway control protocol (SGCP).

4. The system of claim **2** wherein said IP protocol between the terminal proxy and a network server is the H.323 protocol.

5. The system of claim **4** wherein the network server is a PSTN central office switch.

6. The system of claim **4** wherein the network server is a PBX.

7. The system of claim **4** wherein the network server is an H.323 gatekeeper.

8. The system of claim **4** wherein the network server is a SIP proxy.

9. The system of claim **2** wherein said IP protocol between the terminal proxy and a network server is a subset of the Q.931 stimulus protocol.

10. The system of claim **9** wherein the network server is a PSTN central office switch.

11. The system of claim **9** wherein the network server is a PBX.

12. The system of claim **9** wherein the network server is an H.323 gatekeeper.

13. The system of claim **9** wherein the network server is a SIP proxy.

14. A method of providing client proxy signaling in an IP telephony network for a terminal that cannot perform such signaling itself, said method comprising the steps of:

(a) identifying a terminal proxy on an IP network upon initialization of the terminal; and

(b) exchanging call processing messages between the terminal and the terminal proxy such that the terminal proxy manages the call processing logic for the terminal on behalf of the terminal with respect to the remainder of a telephony network.

15. A client proxy signaling system for an IP telephony network comprising:

means for identifying a terminal proxy on an IP network upon initialization of a terminal; and

means for exchanging call processing messages between the terminal and the terminal proxy such that the terminal proxy manages the call processing logic for the terminal on behalf of the terminal with respect to the remainder of a telephony network.

16. A client proxy signaling computer program product for an IP telephony network having a medium with a computer program embodied thereon comprising:

computer program code for identifying a terminal proxy on an IP network upon initialization of a terminal; and

computer program code for exchanging call processing messages between the terminal and the terminal proxy such that the terminal proxy manages the call processing logic for the terminal on behalf of the terminal with respect to the remainder of a telephony network.

* * * * *

EXHIBIT C

US007050861B1

(12) **United States Patent**
Lauzon et al.

(10) **Patent No.: US 7,050,861 B1**
(45) **Date of Patent: May 23, 2006**

(54) **CONTROLLING A DESTINATION TERMINAL FROM AN ORIGINATING TERMINAL**

(75) Inventors: **Eric Lauzon**, Crowthorne (GB); **Bryan J Miller**, Cookham (GB); **Michael O'Doherty**, London (GB)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 1052 days.

(21) Appl. No.: **09/606,053**

(22) Filed: **Jun. 28, 2000**

Related U.S. Application Data

(60) Provisional application No. 60/171,777, filed on Dec. 22, 1999, provisional application No. 60/171,801, filed on Dec. 22, 1999.

(51) **Int. Cl.**
G05G 11/00 (2006.01)

(52) **U.S. Cl.** **700/17**; 700/19; 700/65; 700/83; 709/200; 709/208; 709/220; 709/227; 709/228; 379/188; 379/201.01; 379/201.03; 379/201.05; 379/203.01; 379/204.01

(58) **Field of Classification Search** 700/17, 700/65, 83, 19; 345/804; 709/200, 208, 709/220, 227–228; 379/188, 201.01, 201.03, 379/201.05, 203.01, 204.01
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,867,494 A * 2/1999 Krishnaswamy et al. ... 370/352
6,161,134 A * 12/2000 Wang et al. 709/220
6,285,364 B1 * 9/2001 Giordano et al. 345/804
6,359,892 B1 * 3/2002 Szlam 370/401
6,446,070 B1 * 9/2002 Arnold et al. 707/10
6,446,127 B1 * 9/2002 Schuster et al. 709/227
6,502,127 B1 * 12/2002 Edwards et al. 709/206
6,564,261 B1 * 5/2003 Gudjonsson et al. 709/227
6,584,490 B1 * 6/2003 Schuster et al. 709/200
6,615,236 B1 * 9/2003 Donovan et al. 709/203
6,714,536 B1 * 3/2004 Dowling 370/356

OTHER PUBLICATIONS

SIP : Session Initiation Protocol; RFC 2543; by Handley et al; published Mar. 1999; entire document.*
ChaiTime: A System for Rapid Creation of Portable Next-Generation Telephony Services Using Third-Party Software Components, Farooq Anjum et al., IEEE Mar. 26-27, 1999; pp. 22-31.*
"Under the Hood: The architecture of aglets", by Bill Venners, JavaWorld Apr. 1997.

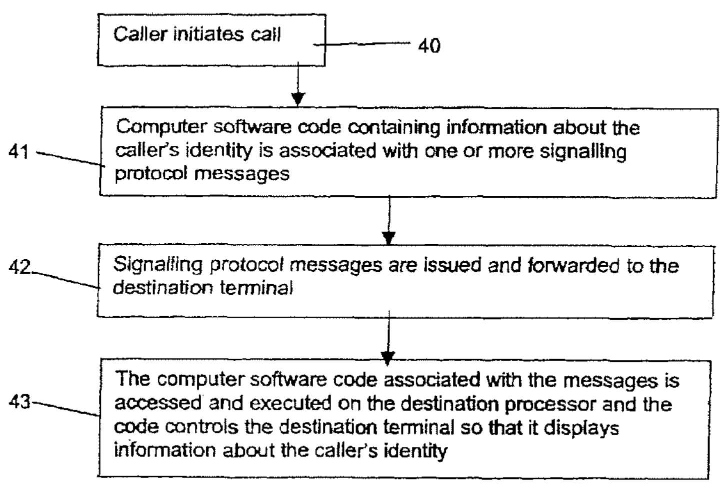
(Continued)

Primary Examiner—Anthony Knight
Assistant Examiner—Ronald D Hartman, Jr.
(74) *Attorney, Agent, or Firm*—Barnes & ThornburgLLP

(57) **ABSTRACT**

A caller associates computer software code with a signalling protocol messages such that when the messages are received at a destination processor the computer software code is executed. For example, the messages may be improved SIP protocol messages with incorporated Java code. By selecting different computer software code for association with the messages, the caller is able to control the destination terminal. For example, to display information about the identity of the caller at the destination terminal; to modify the behaviour of the destination terminal according to the priority of the call; to take into account the configuration of the destination terminal, and to allow users to adjust this configuration from a remote location.

13 Claims, 11 Drawing Sheets



OTHER PUBLICATIONS

Krause, "Mobile Service Agents Enabling "Intelligence on demand" in Telecommunications", Global Telecommunications Conference 1996, pp 78-84.
Kristensen, "The SIP Servlet API", Internet Draft, 1999, pp 1-26.
Kaukonen, "Agent-Based Conferencing Using Mobile IP-Telephony", 3rd Workshop on Multimedia Signal Processing, Sep. 1999, pp 593-598.
Guedes, "An Agent-Based Approach for Supporting Quality of Service In Distributed Multimedia Systems", Computer Communicaiton, Sep. 15, 1998, pp. 1269-1278.
Breugst, "Mobile agents--enabling technology for active intelligent network implementation," May-Jun., 1998, 8 pages.
"A Beginner's Guide to URL's", National Center for Supercomputing Applications, Dec. 16, 2003.

* cited by examiner

PRIOR ART

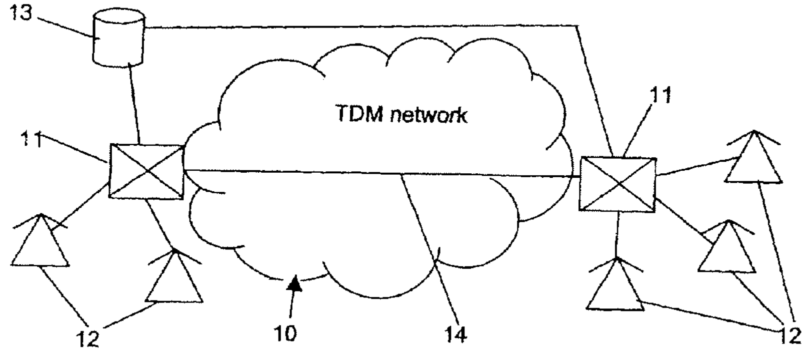


Figure 1

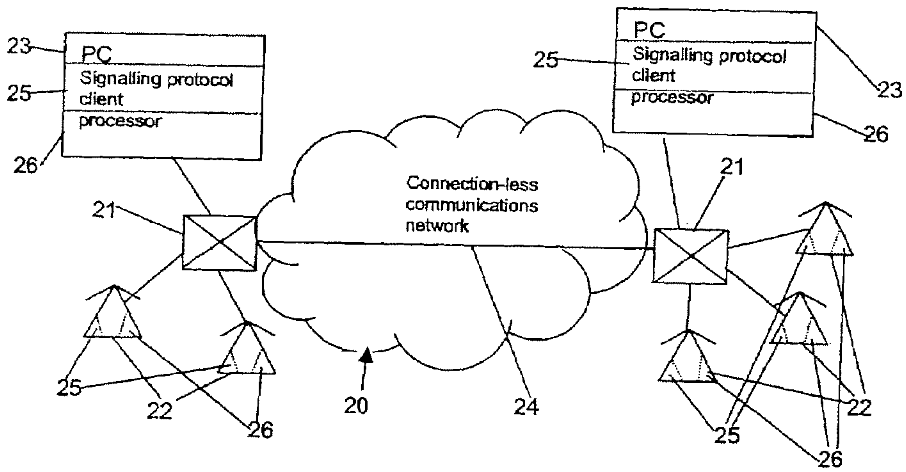


Figure 2

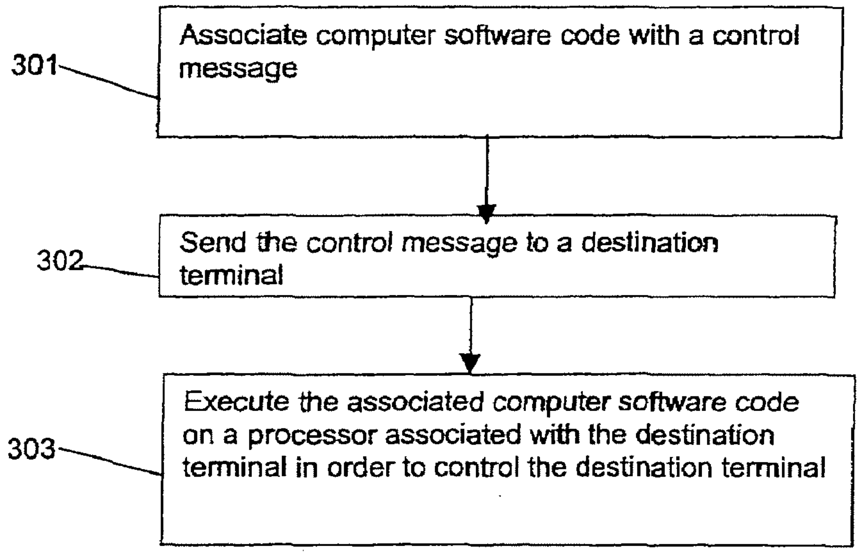


Figure 3

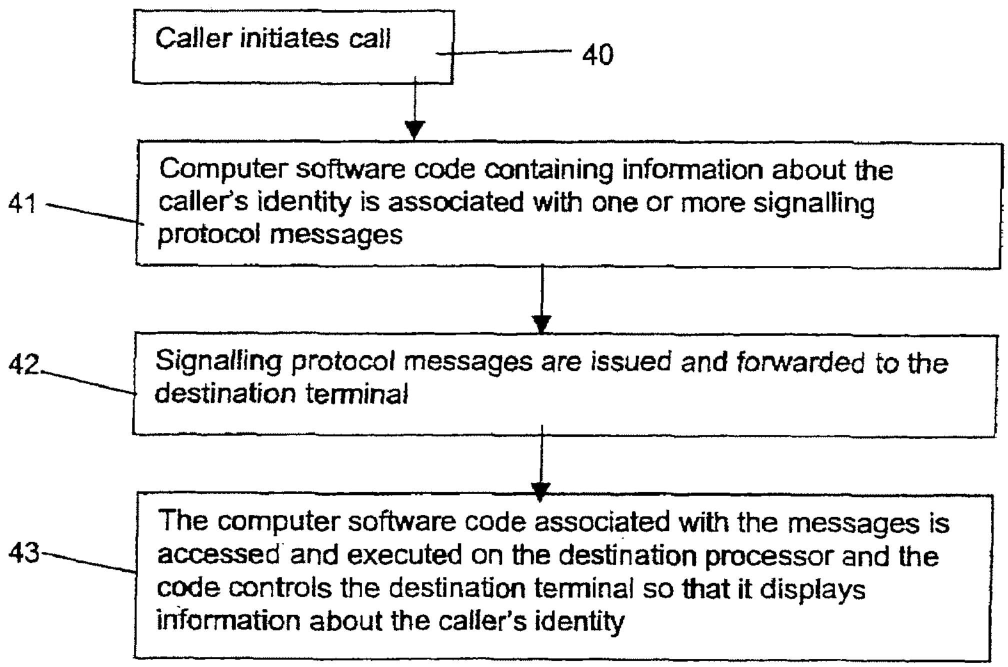


Figure 4

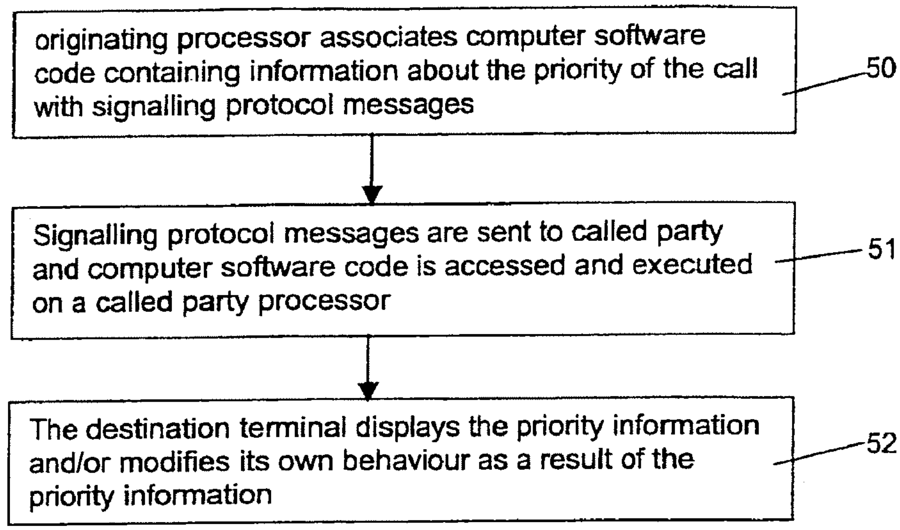


Figure 5

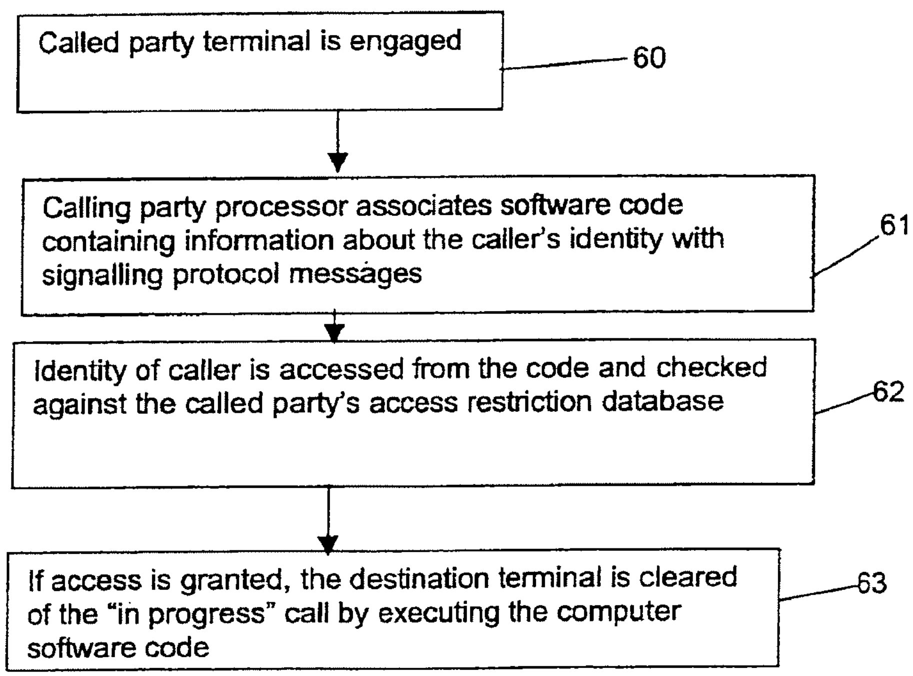


Figure 6

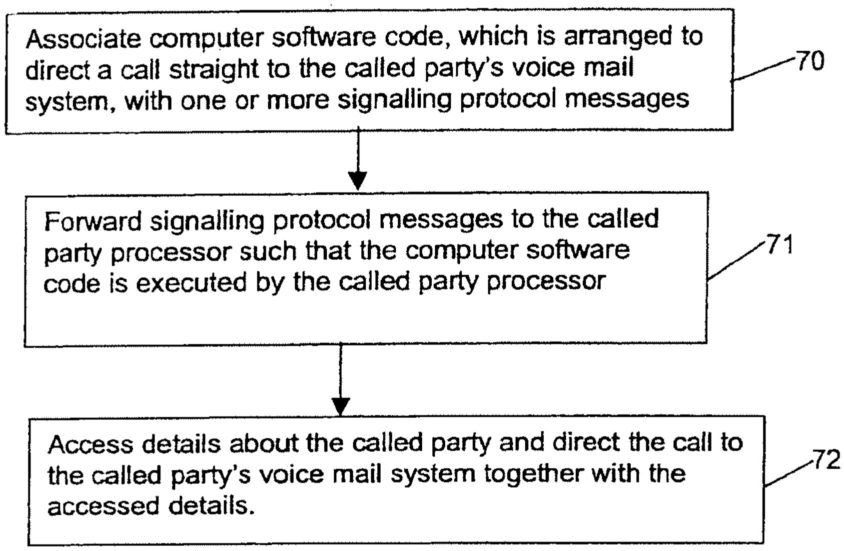


Figure 7

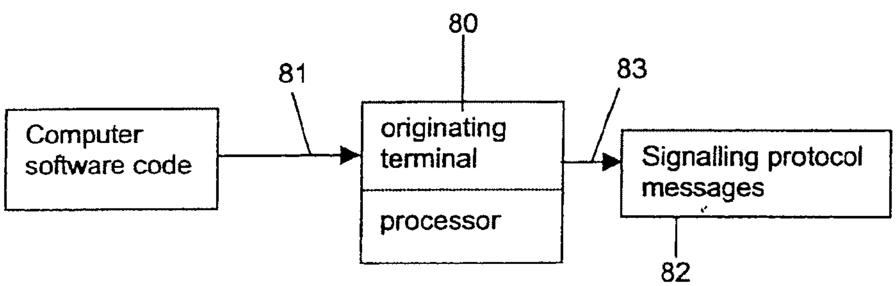


Figure 8

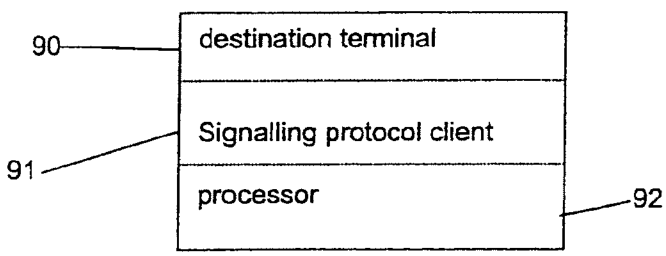


Figure 9

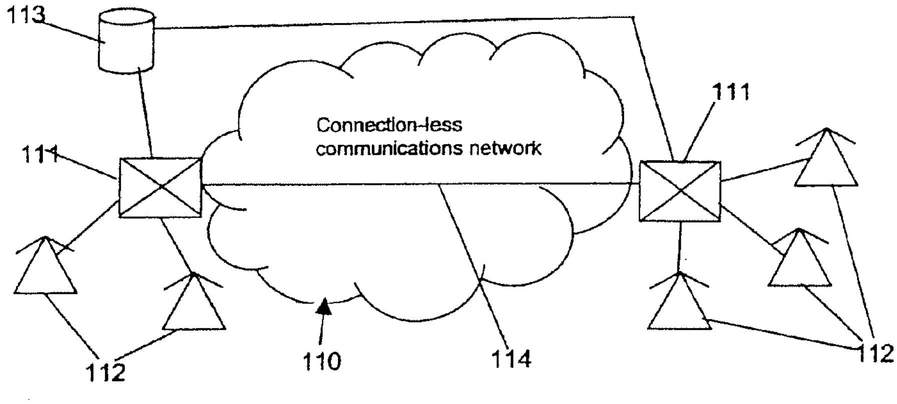


Figure 10

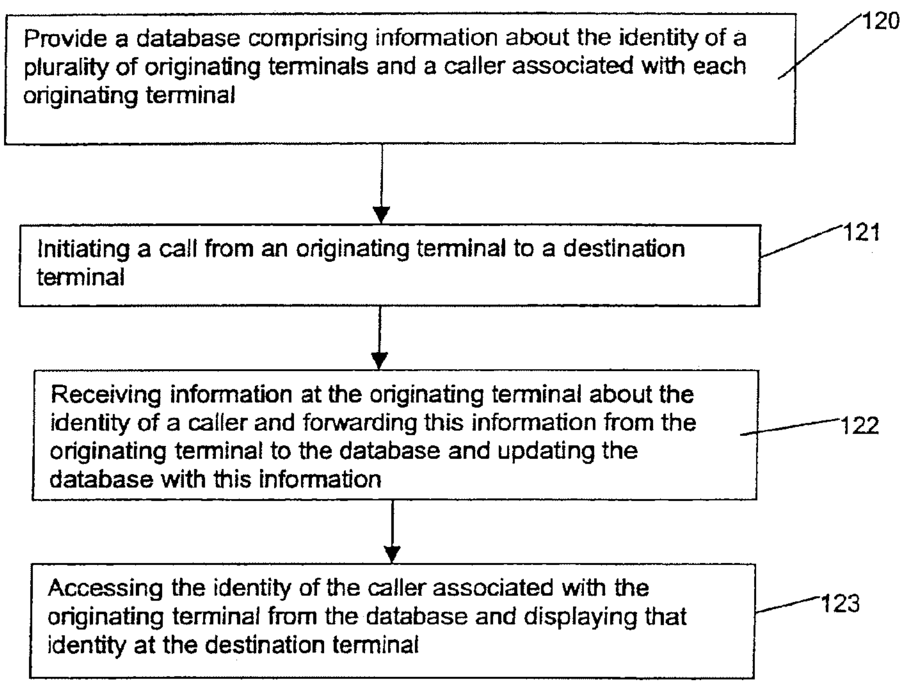


Figure 11

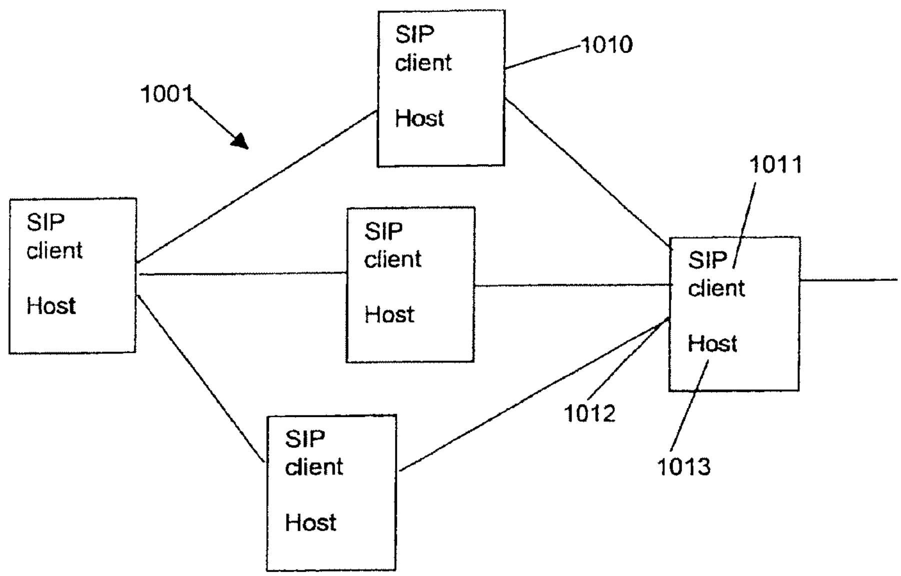


Figure 12

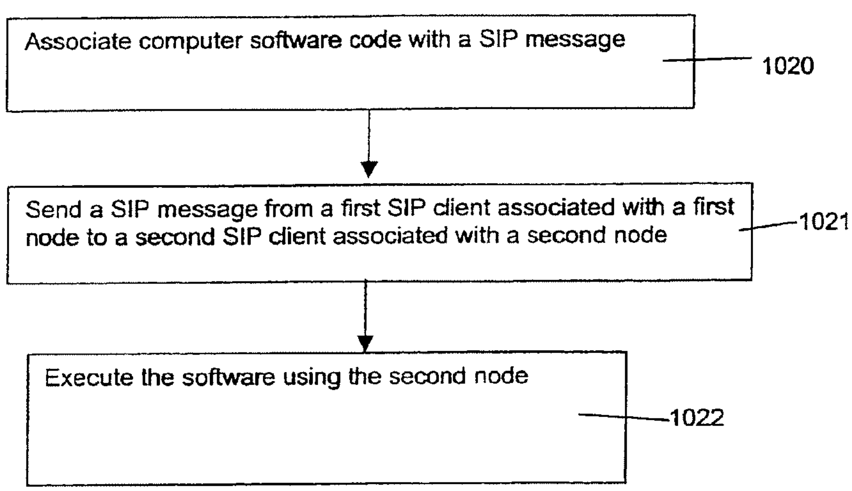
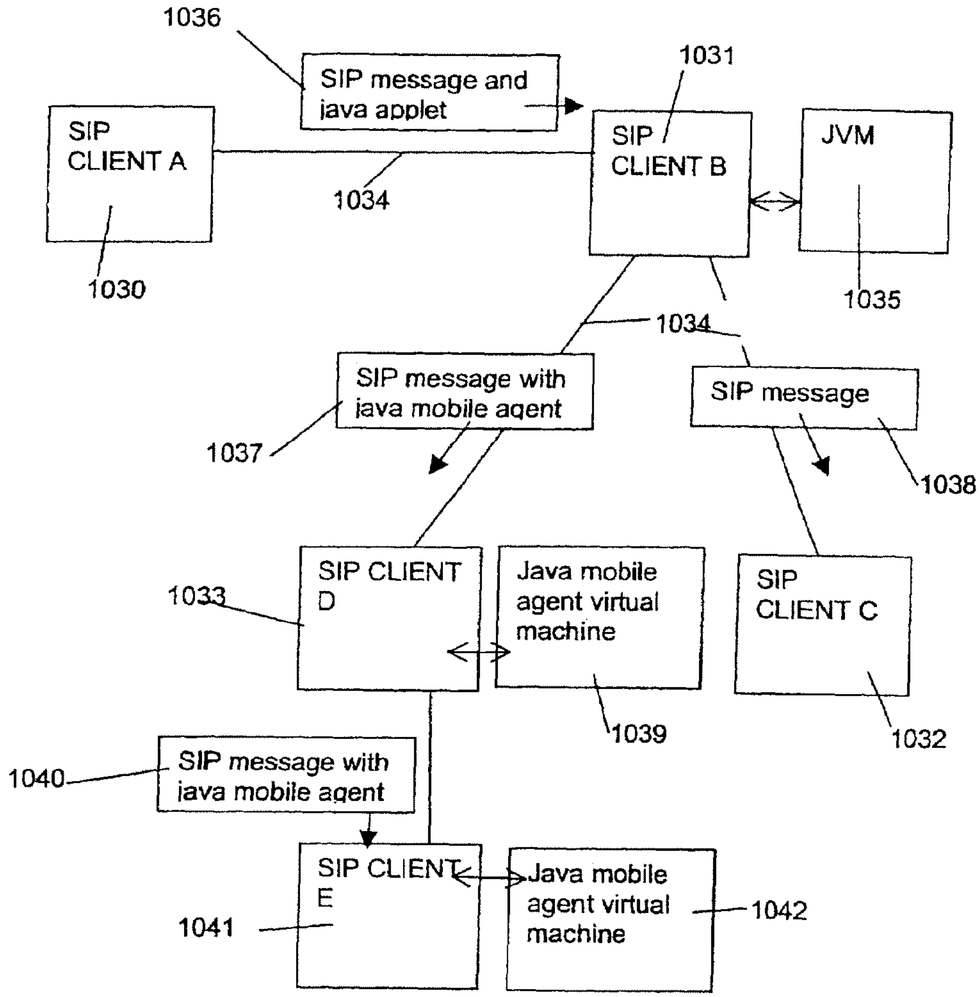


Figure 13



1036
SIP message and java applet
1031
SIP CLIENT A
SIP CLIENT B
JVM
1034
1030
1035
1034
SIP message with java mobile agent
SIP message
1037
1038
SIP CLIENT D
Java mobile agent virtual machine
SIP CLIENT C
1033
1039
1032
1040
SIP message with java mobile agent
SIP CLIENT E
Java mobile agent virtual machine
1041
1042

Figure 14

C ->S: INVITE sip:watson@boston.bell-tel.com SIP/2.0
Via: SIP/2.0/UDP kton.bell-tel.com>
From: A. Bell ,sip:a.g.bell@bell-tel.com>
To: T. Watson ,sip:watson@bell-tel.com.
Call-ID: 3298420296@kton.bell.tel.com
Cseq: 1 INVITE
Subject: Mr. Watson, come here.
Content-Type: multipart/mixed; boundary=3E4A567F4C8A
(or URL for java applet)
Content-Length: ...
Require: org.ietf.sip.java-enhanced-sip

(Within the message body)

--3E4A567F4C8A
Content-Type: application/x-sipjava
Content-Encoding: binary
Content-length: xxx
...Java applet or Java mobile agent for SIP message processing...
--3E4A567F4C8A--

1050

Figure 15

```
Request    =  Request-Line                        }
              *( general-header ;                 }
               | request-header                   }  1040
               | entity-header                    }
              CRLF                                }
              [ message-body ]  ;                 }

general-header   = Accept                    ;   }
                 | Accept-Encoding           ;   }
                 | Accept-Language           ;   }
                 | Call-ID                   ;   }
                 | Contact                   ;   }
                 | CSeq                      ;   }
                 | Date                      ;   }  1041
                 | Encryption                ;   }
                 | Expires                   ;   }
                 | From                      ;   }
                 | Record-Route              ;   }
                 | Timestamp                 ;   }
                 | To                        ;   }
                 | Via                       ;   }
     { entity-header   = Content-Encoding    ;
1042 {                 | Content-Length      ;
     {                 | Content-Type  <------;- This will indicate that
     { request-header  = Authorization       ;  the content type is a
     {                 | Contact             ;  java applet or a Java
     {                 | Hide                ;  Mobile Agent (or the URL
     {                 | Max-Forwards        ;  of a location of either
     {                 | Organization        ;  from where they must be
     {                 | Priority            ;  retrieved).
1043 {                 | Proxy-Authorization ;
     {                 | Proxy-Require       ;
     {                 | Route               ;- This will be used to
     {                 | Require  <----------;  indicate that Java-
     {                 | Response-Key        ;  enhanced-SIP must be
     {                 | Subject             ;  supported to process
     {                 | User-Agent          ;  this message
response-header  = Allow                     ;   }
                 | Proxy-Authenticate        ;   }
                 | Retry-After               ;   }
                 | Server                    ;   }  1044
                 | Unsupported               ;   }
                 | Warning                   ;   }
                 | WWW-Authenticate          ;   }
```

Figure 16

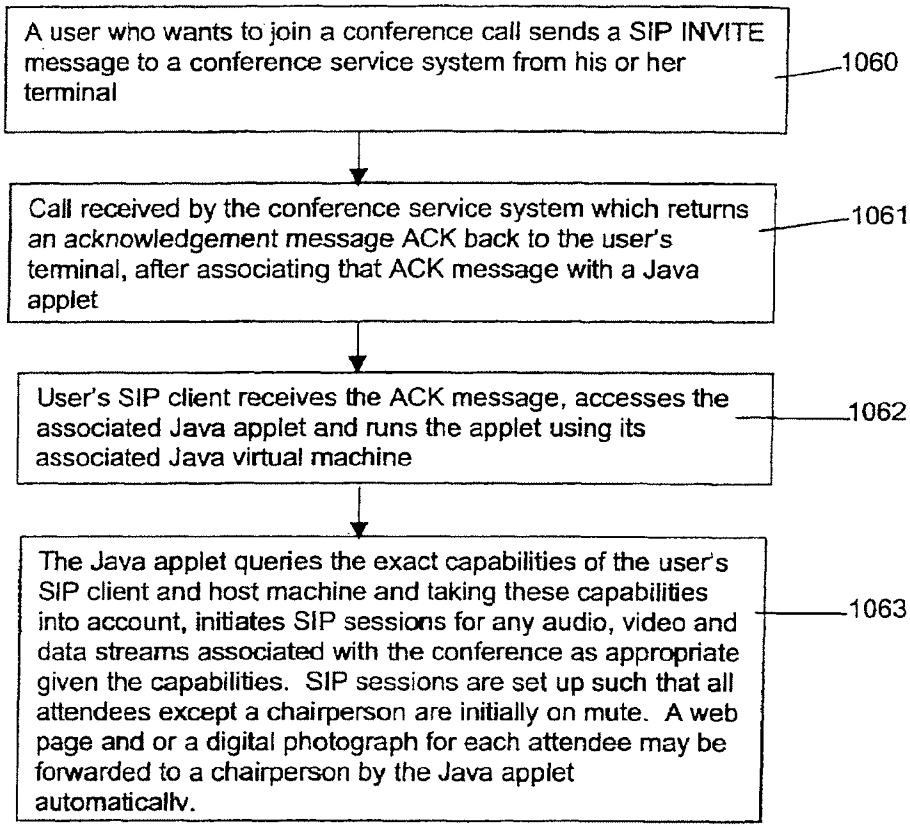


Figure 17

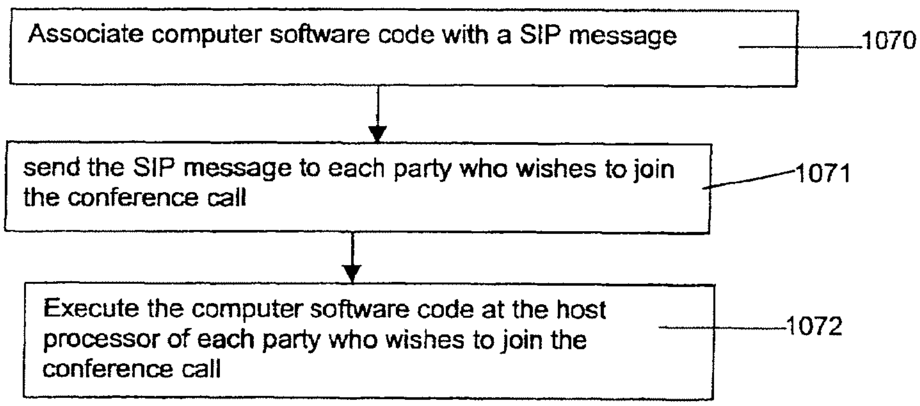


Figure 18

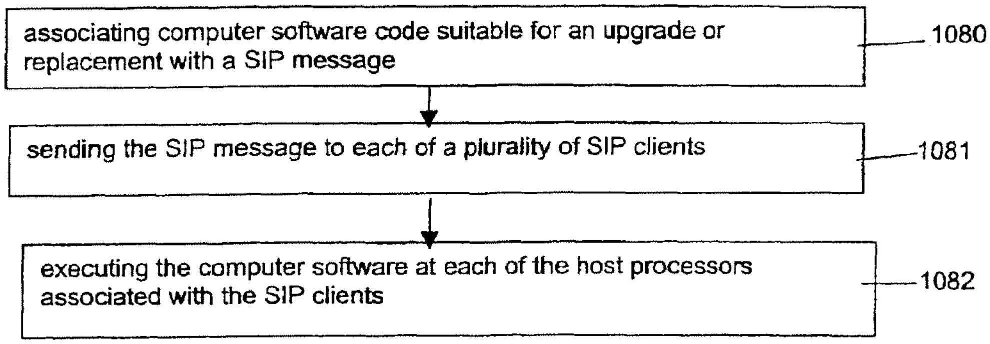


Figure 19

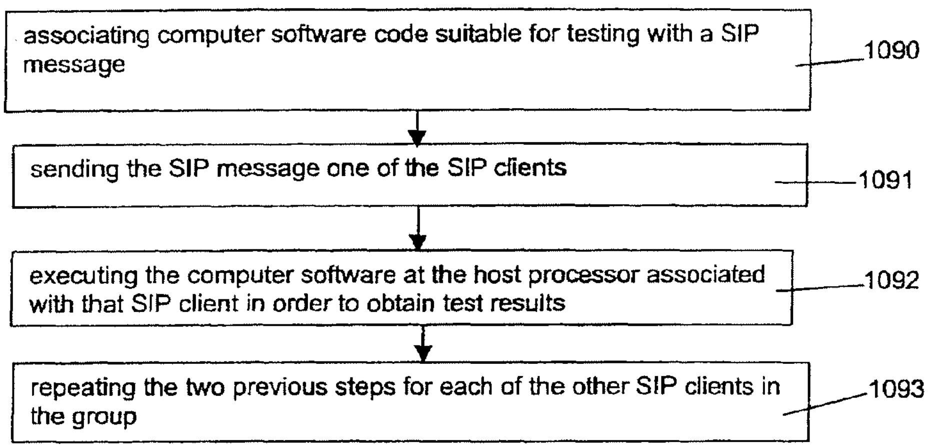


Figure 20

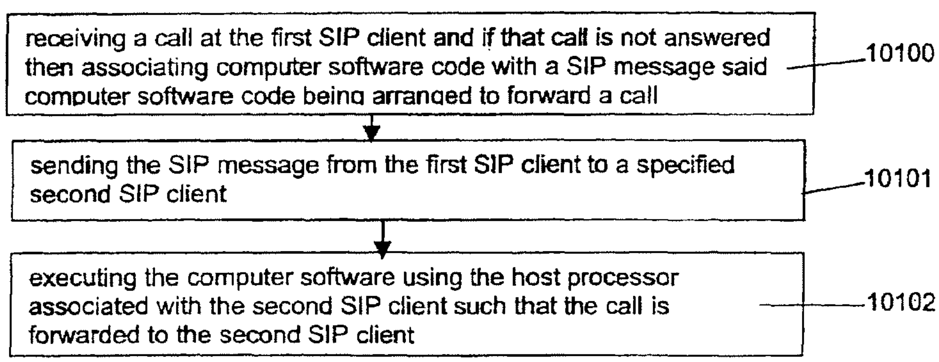


Figure 21

CONTROLLING A DESTINATION TERMINAL FROM AN ORIGINATING TERMINAL

RELATED APPLICATIONS

This application is the non-provisional filing of provisional U.S. Patent Applications 60/171,777 and 60/171,801, both filed on Dec. 22, 1999.

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates to a method of remotely controlling a destination terminal from an originating terminal. The invention is particularly related, but in no way limited to, using improved session initiation protocol (SIP) to enable a caller to control an originating terminal.

2. Description of the Prior Art

The amount of control that an originating terminal has over a destination terminal has been very restricted. For example, when making an extremely urgent call to a busy destination terminal, the caller is unable to free up the busy destination terminal by causing the call that is currently in progress to be dropped. Also, the called party may have particular services set-up on his or her terminal and the calling party is unable to take these into account easily or to modify the set-up services. This is particularly problematic when a caller wishes to adapt his or her call as a result of taking the called party's terminal configuration into account. For example, a user may be accustomed to setting his or her terminal to ring three times before going to voice mail, during times when that user is resting. At other times, suppose that the user sets his or her terminal to ring five times before going to voice mail. The user's family members may wish only to make a call to the user when the user is not resting. However, this is not possible because callers are unable to take into account set-up configurations on the user's terminal.

Similarly, calling parties are unable to easily provide information to the called party and to cause the destination terminal to display or act upon this information. For example, a calling party may wish to provide information about his or her identity to the called party. In the past this has been done by associating each terminal with a particular user. However, this is problematic when users move about and use different terminals. Also, prior art systems which display the caller identity at the destination terminal are fixed systems. That is, the caller is unable to easily change or modify the manner in which the destination terminal displays or acts upon the identity information.

It is accordingly an object of the present invention to provide a method of remotely controlling a destination terminal from an originating terminal, which overcomes or at least mitigates one or more of the problems noted above.

SUMMARY OF THE INVENTION

According to an aspect of the present invention there is provided a method of remotely controlling a destination terminal from an originating terminal said destination terminal having an associated signalling protocol client and an associated processor comprising the steps of:

- associating computer software code with at least one signalling protocol message;
- sending the signalling protocol message to the destination terminal from the originating terminal;
- executing the computer software code using the processor associated with the destination terminal in order that the originating terminal controls the destination terminal.

This provides the advantage that an originating terminal is able to control a destination terminal. For example, to display information about the identity of the caller on the destination terminal or to modify the behaviour of the destination terminal on the basis of priority information provided by the calling party.

According to another aspect of the present invention there is provided an originating terminal arranged to control a destination terminal said originating terminal comprising:—

- an input arranged to access computer software code suitable for controlling said originating terminal;
- a processor arranged to associate said computer software code in use with one or more signalling protocol messages; and
- an output arranged to route said signalling protocol messages to the destination terminal in use.

This provides the advantage that by using such an originating terminal a user is able to control a destination terminal.

According to another aspect of the present invention there is provided a destination terminal comprising:—

- a signalling protocol client arranged to receive one or more signalling protocol messages sent from an originating terminal;
- a processor arranged to access any computer software code associated with received signalling protocol messages in use; and wherein said processor is arranged to execute such accessed computer software code such that the destination terminal is controlled.

This provides the advantage that a destination terminal which can be controlled from a remote location by an originating terminal is provided.

According to another aspect of the present invention there is provided a signal comprising one or more signalling protocol messages which are associated with computer software code. This provides the advantage that the functions of the signalling protocol messages are greatly extended. For example, the signalling protocol messages can be sent from an originating terminal to a destination terminal to control that destination terminal.

According to another aspect of the present invention there is provided a method of displaying information about the identity of a caller at a destination terminal comprising the steps of:

- providing a database comprising information about the identity of a plurality of originating terminals and a caller associated with each originating terminal;
- initiating a call from an originating terminal to a destination terminal;
- receiving information at the originating terminal about the identity of a caller and forwarding this information from the originating terminal to the database and updating the database with this information; and
- accessing the identity of the caller associated with the originating terminal from the database and displaying that identity at the destination terminal.

This provides the advantage that the database of caller identity information is updated prior to use so that the identity information displayed is correct, even if the caller uses different terminals or several users use the same terminal.

Further benefits and advantages of the invention will become apparent from a consideration of the following

detailed description given with reference to the accompanying drawings, which specify and show preferred embodiments of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic diagram of a time division multiplex (TDM) communications network arrangement according to the prior art.

FIG. 2 is a schematic diagram of a connectionless communications network suitable for use with the present invention.

FIG. 3 is a flow diagram of a method of controlling a destination terminal from an originating terminal.

FIG. 4 is a flow diagram of a method of controlling a destination terminal such that information about the identity of the calling party is displayed at the destination terminal.

FIG. 5 is a flow diagram of a method of controlling a destination terminal using information about the priority of the call.

FIG. 6 is a flow diagram of a method of controlling a destination terminal in order to clear an "in progress" call from the destination terminal.

FIG. 7 is a flow diagram of a method for controlling a destination terminal in order that the call is directed straight to a voice mail system.

FIG. 8 is a schematic diagram of an originating terminal.

FIG. 9 is a schematic diagram of a destination terminal.

FIG. 10 is a schematic diagram of a connectionless communications network suitable for use with an embodiment of the present invention

FIG. 11 is a flow diagram of a method of displaying information about the identity of a caller at a destination terminal.

FIG. 12 is a schematic diagram of a communications network which incorporates nodes for implementing an improved SIP protocol.

FIG. 13 is a flow diagram of a method of communicating between two SIP clients using an improved SIP protocol.

FIG. 14 is a schematic diagram of interaction between a plurality of SIP clients according to the improved SIP protocol.

FIG. 15 shows the format of an improved SIP protocol message.

FIG. 16 is an example of an improved SIP protocol INVITE message.

FIG. 17 is a flow diagram of a method of setting up a conference call using a conference call service system.

FIG. 18 is a flow diagram of a method of setting up a conference call.

FIG. 19 shows a method of upgrading or replacing interconnected SIP clients.

FIG. 20 shows a method of testing members of a group of SIP clients.

FIG. 21 shows a method of forwarding a call from a first SIP client to a second SIP client.

DETAILED DESCRIPTION OF THE INVENTION

Embodiments of the present invention are described below by way of example only. These examples represent the best ways of putting the invention into practice that are currently known to the Applicant although they are not the only ways in which this could be achieved.

The term "originating terminal" is used to refer to an apparatus via which a user is able to send communications into a communications network in order to call another party; for example, a telephone handset, a computer terminal or a mobile telephone handset.

The term "destination terminal" is used to refer to an apparatus via which a user is able to receive communications from the communications network in order to be called by another party; for example, a telephone handset, a computer terminal or a mobile telephone handset.

The term "calling party" is used to refer to an entity which sends a communication into a communications network in order to communicate with a called party.

The term "called party" is used to refer to an entity which receives communications from a calling party via a communications network.

The present application is at least in part an extension of Nortel Networks's earlier work described in co-assigned, earlier U.S. patent application Ser. No. 09/520,853, filed on 7 Mar. 2000 (Nortel reference 11790 ID). That patent document describes an improved Session Initiation Protocol (SIP). Using this improved SIP protocol computer software code is associated with SIP messages. These SIP messages are sent to a SIP client which is arranged to execute the software code associated with the SIP messages. The specific description from U.S. patent application Ser. No. 09/520,853 is repeated in Appendix A.

FIG. 1 shows a prior art arrangement in which a plurality of terminals 12 (such as telephone handsets) are connected to a time division multiplex (TDM) communications network (such as a public switched telephone network) via access nodes 11. A database 13 is also provided which is accessible by each of the access nodes 12. The database contains pre-specified information about the identity of each terminal 12 (for example, the calling line identifier (CLID)) and the name of a user associated with each terminal. When a caller initiates a call, the name associated with the terminal from which the call is being made is accessed from the database 13 and displayed at the called terminal. In some circumstances this lets the called party know who is calling before the call is answered. However, often one particular terminal is associated with more than one person and in addition, callers are mobile and often use different terminals to make calls. Arrangements like that illustrated in FIG. 1 are not able to deal with these situations and simply display the name of the one user associated with the particular terminal being used, even if a different person is actually using that terminal.

Another prior art arrangement involves storing pre-specified information about the identity of terminals at their associated access nodes 11. For example, this information comprises the CLID of each terminal 12 which is connected to the access node 11 and the name of a user associated with each of those terminals 12. When a caller initiates a call, the name associated with the terminal from which the call is being made is sent with the call to the destination terminal. The name information is static and because of this the system is not flexible and cannot take account of the fact that different users use the same originating terminal or that individual users move about and use different terminals.

FIG. 2 illustrates an embodiment of the present invention in which information about the identity of a caller is made available to the called party independently of the particular terminal being used by the caller. A plurality of terminals 22, 23 are connected to a connectionless communications network 20 such as an internet protocol (IP) communications network via access nodes 21 such as voice over internet protocol (VoIP) gateways. Calls are set-up between two terminals using any suitable signalling protocol such as

session initiation protocol (SIP). The terminals may be for example, personal computer based telephones **23** or conventional telephone handsets **22**. Associated with each terminal is a signalling protocol client **25** which is a computer program that is arranged to control the terminal such that it is able to send and/or receive messages according to the particular signalling protocol being used. This signalling protocol client program **25** may be provided on any suitable computing platform integral with the terminal or accessible by the terminal. As well as the signalling protocol client **25**, a processor **26** is associated with each terminal and the processor **26** is arranged to execute any computer software code that is associated with signalling protocol messages received from callers.

With reference to FIG. **4**, when a caller initiates a call (box **40** of FIG. **4**), computer software code is associated with one or more signalling protocol messages issued by the caller's terminal in order to set-up the call (box **41** of FIG. **4**). This computer software code contains information about the caller's identity or a reference to this information. The signalling protocol message issued by the caller's terminal is forwarded (box **42** of FIG. **4**) to the called party's terminal and the associated computer software code is accessed. This code is then executed on the processor **26** associated with the called party's terminal, provided that security provisions on the destination terminal allow this (box **43** of FIG. **4**). The executed code controls the destination terminal such that it displays the identity of the caller. For example, by playing a sample of the caller's voice or by displaying the caller's name on a visual display.

By using this method, the caller's identity is correct no matter which terminal the caller uses and it is not necessary to make use of CLID information.

The example described above of allowing a caller to control a destination terminal in order to display information about the caller's identity is only one embodiment of the present invention. More generally, the invention provides a way for callers to control a destination terminal by selecting computer software code for association with SIP messages, or any other suitable signalling protocol messages. This control of the destination terminal is of course subject to any security and access restriction arrangements that are set-up on the destination terminal. The default situation is that the destination terminal is controlled by its associated signalling protocol client and associated processor. Thus the caller does not have absolute control over the destination terminal except in cases where the security and access restrictions allow this.

A calling party (or other user) is able to select or create the computer software that is to be associated with the signalling protocol messages using a user interface such as a graphical user interface (GUI). Once this code is selected or created it is stored in a location that is accessible by the calling party's signalling protocol client. This location could be at the terminal itself, at a gateway from which the terminal subtends or at any other suitable location. In addition, rules or other criteria are stored which specify when particular pieces of the stored computer software are to be associated with SIP or other signalling protocol messages.

FIG. **3** is a flow diagram for a method of controlling a destination terminal from an originating terminal. Computer software code is first associated with a signalling protocol message (box **301**) and then that signalling protocol message is forwarded to a destination terminal (see box **302** of FIG. **3**). The computer software code is then executed on a processor associated with the destination terminal in order to control the destination terminal (box **303** of FIG. **3**).

The computer software code may be associated with the signalling protocol message in any suitable manner, for example, by adding the code to the message or adding a reference to the location of the code to the message. Any suitable signalling protocol messages may be used, such as session initiation protocol (SIP) messages. Appendix A gives more details about this.

Several different examples of ways in which the destination terminal is controlled are now described.

In one example, the caller is able to provide information about the priority of the call. In the past this has not been possible for conventional public switched telephone network systems where the CLID and ringing tone are all that is available to alert the called party to the call request. Answering machines can be used but in that case the called party must be available to listen to incoming calls and answer these if they are urgent.

FIG. **5** is a flow diagram of a method of controlling a destination terminal using information about the priority of the call. The calling party processor associates computer software code which contains information about the priority of the call (or a reference to the location of this information) with one or more signalling protocol messages (box **50** of FIG. **5**). When the signalling protocol message is received by the destination terminal, the code is executed as described above (box **51** of FIG. **5**) and this causes the priority information to be displayed and/or to affect the behaviour of the destination terminal (box **52** of FIG. **5**). For example, if the priority of the call is very urgent, then the code may cause the destination terminal to re-direct the call to an associated mobile telephone. As mentioned above this is subject to access and security restrictions. For example, the called party may have set up a database containing the identities of callers who should be given access at all times, those to whom access is to be denied and those to whom access should be given only during certain time periods. In this case, information about the caller's identity is obtained and used to determine which access levels are to be given.

In some situations, the destination terminal is engaged. In this case the computer software code associated with the signalling protocol message may be arranged to cause the destination terminal to be cleared (subject to security and access restrictions). For example, the caller may be trying to reach a family member urgently. The called party has previously stored the names of people who are allowed to cause the called party's terminal to be cleared of an "in progress" call. The called party may also have set up a password system whereby the caller must provide the password before being able to clear "in progress" calls. In this case, the computer software code sent by the caller with the signalling protocol message contains the password and/or name of the caller.

This process is illustrated in more detail in FIG. **6**. In the situation when the destination terminal is engaged (box **60** of FIG. **6**) the calling party processor associates computer software code, containing information about the caller's identity and code for clearing the "in progress" call from the destination terminal, with signalling protocol messages (box **61** of FIG. **6**). These messages are forwarded to the called party and the called party processor accesses the information about the identity of the caller (box **62** of FIG. **6**). The called party processor checks the identity of the caller against an access restriction database (previously set up by the called party). If access is granted to the particular caller, then the software code is executed in order to clear the "in progress" call from the destination terminal (box **63** of FIG. **6**).

The called party may also configure the signal processing client associated with its terminal such that "in progress" calls can only be cleared under certain circumstances. For example, when the "in progress" call is to an internet service provider or to one of a list of pre-specified destinations. In this way the called party is able to specify things like "If I am using the internet I am happy to allow family members to shut down my internet connection in order that they can telephone me." The called party is able to set up the security and access restrictions by using a user interface to modify the signal protocol client and any other software which controls the processor associated with the destination terminal.

The calling party is also able to select or create appropriate computer software code such that the configuration of the destination terminal is checked and taken into account before taking further action. For example, the number of rings at the destination terminal may be set to 3 before the call is diverted to a voice mail system. A caller may know that the called party only sets this number of rings to three when he or she is resting. In that case, the caller may prefer not to disturb the called party at all. The caller is then able to arrange the computer software code associated with the signalling protocol message such that it checks the "number of rings before divert to voice mail" setting at the destination terminal before proceeding with the call.

Known telephone systems, such as those in North America, have a facility whereby the calling party is able to block information about the CLID from the called party. This facility is often used by mobile phone callers who wish to prevent others from obtaining their mobile phone number. This is because they wish to prevent others from making calls to their mobile telephone which incur cost to the mobile phone owner. However, many non-mobile phone users have made use of the blocking facility, for example, sales people who wish to hide their identity in order that people will answer their calls. This has led to the creation of a service by which users are able to "block the blocker"; that is, users are able to block calls from any party who has blocked information about their identity from being made available to the called party. This "block the blocker" facility can be problematic in some circumstances. For example, consider a mobile phone user who has made use of the blocking facility. If that mobile phone user makes a call to a family member that family member is unaware of the identity of the caller. Suppose that the family member has implemented the "block the blocker" function on his or her terminal. In that case the user's call to the family member is blocked, even though that call may be extremely urgent. By making use of the present invention this problem is overcome. The user is able to control the family member's terminal in order to override the "block the blocker" function. For example, the caller sends signalling protocol messages containing a password which the called party receives and checks against pre-specified security criteria. If security clearance is obtained, software code associated with the signalling protocol messages ensures that the "block the blocker" function on the destination terminal is over-ridden.

In another example, the caller is able to control the destination terminal to give preferred handling to the call. For example, the caller is able to control the destination terminal such that the call is directed straight to a voice mail service or straight to the called party's mobile telephone. Prior art systems which allow a user to call a voice mail system directly (rather than being diverted to the voice mail system from the destination terminal) are difficult to use. Typically the caller must dial the number to connect to the voice mail system and then enter details about who is being called. This is time consuming and complex. By using the present invention this problem is avoided because a call with the called party is actually established unlike the prior art situation where a call is established directly with the voice mail system.

FIG. **7** is a flow diagram of a method for controlling a destination terminal in order that the call is directed straight to a voice mail system. The calling party processor associates computer software code with one or more signalling protocol messages (box **70**, FIG. **7**). This computer software code is arranged to control the destination terminal such that the call is directed straight to the voice mail system rather than causing the destination terminal to ring. The signalling protocol messages are forwarded to the called party (box **71** FIG. **7**) and the computer software code accessed and executed on the called party processor (subject to any security restrictions). A call is effectively established between the calling and called party at this stage, although the destination terminal does not ring. Because a call is effectively established, details about the called party are available. The computer software code accesses these details and controls the called party processor such that the call is directed to the called party's voice mail system. The information about the called party is also forwarded to that voice mail system so that the called party is not required to re-enter these details (box **72** of FIG. **7**).

In another example, a user is able to adjust the configuration of his or her terminal from a remote location. For example, that user acts as a calling party and calls his or her own terminal. Using the method described herein for controlling destination terminals, the user is then able to control his or her own terminal. For example, the user is able to adjust services such as "number of rings before call sent to voice mail" and other such terminating services from a remote location. This is achieved by associating appropriate computer software code with signalling protocol messages and forwarding these to the called party processor for execution.

FIG. **8** shows an originating terminal **80** in more detail. The originating terminal has:

- an input **81** arranged to access computer software code suitable for controlling said originating terminal;
- a processor **82** arranged to associate said computer software code in use with one or more signalling protocol messages; and
- an output **83** arranged to route said signalling protocol messages to the destination terminal in use.

It is not essential for the processor **82** to be integral with the originating terminal **80**. It is also possible for the processor to be physically separate from the originating terminal as long as communication between the processor and originating terminal is provided.

FIG. **9** shows a destination terminal **90** in more detail. The destination terminal **90** comprises:

- a signalling protocol client **91** arranged to receive one or more signalling protocol messages sent from an originating terminal;
- a processor **92** arranged to access any computer software code associated with received signalling protocol messages in use; and wherein said processor is arranged to execute such accessed computer software code such that the destination terminal is controlled.

As for the originating terminal, it is not essential for the processor **92** to be integral with the destination terminal **90**. The same applies for the signalling protocol client **91**.

However, communication between the processor **92** and the destination terminal **90** and between the signalling protocol client **91** and the destination terminal **90** must be provided.

FIG. **10** illustrates another embodiment of the present invention. A plurality of terminals **112** (such as telephone handsets) are connected to a connectionless communications network (such as an internet protocol communications network) via access nodes **111**. A database **113** is also provided which is accessible by each of the access nodes **112**. The database contains pre-specified information about the identity of each terminal **112** (for example, the calling line identifier (CLID)) and the name of a user associated with each terminal. When a caller initiates a call, information about the caller's identity is forwarded to the database **113** and used to update the database **113**. For example, the identity information is forwarded to the database **113** by being associated with a signalling protocol message that is forwarded to the database. Also, it is not essential for the database to be located separately from other components of the communications network. For example, the database may be incorporated into the access nodes **111**. The caller's identity is then accessed from the database **113** by a destination terminal and displayed at that destination terminal. By dynamically updating the database **113** in this way, the correct identity information is displayed no matter whether the user uses different terminals **112** or several users use the same terminal.

FIG. **11** is a flow diagram of a method of displaying information about the identity of a caller at a destination terminal comprising the steps of:

providing a database comprising information about the identity of a plurality of originating terminals and a caller associated with each originating terminal (box **120**);

initiating a call from an originating terminal to a destination terminal (box **121**);

receiving information at the originating terminal about the identity of a caller and forwarding this information from the originating terminal to the database and updating the database with this information (box **122**); and

accessing the identity of the caller associated with the originating terminal from the database and displaying that identity at the destination terminal (box **123**).

A range of applications are within the scope of the invention. These include situations in which it is required to control a destination terminal from an originating terminal. For example, to cause information about the identity of a caller to be displayed at the destination terminal. Another example involves providing information about the priority of a call and allowing the behaviour of the destination terminal to be adjusted in response to the call priority. As well as this, it is possible to clear an "in progress" call from an engaged destination terminal and to take into account configuration information on the destination terminal. Users are also able to adjust the configuration of terminating services on their terminals from a remote location.

APPENDIX A

A method of associating computer software code with signalling protocol messages such as Session Initiation Protocol (SIP) messages is now described by repeating some of the text from Nortel Network's earlier co-assigned U.S. patent application Ser. No. 09/520,853. However, it is not essential to use the improved SIP protocol described below. Any suitable protocol and method for associating computer software code with signalling protocol messages may be used.

The term "SIP Client" is used to refer to a computer program that is arranged to control a communications network node such that it is able to send SIP messages such as SIP request messages. The computing platform that the SIP client runs on is referred to as a "host system". The communications network node either comprises the host system or is associated with the host system.

The term "Java virtual machine" is used to refer to a processor which is arranged to execute Java applets or Java byte code.

The term "mobile autonomous software agent" is used to refer to a computer program that is able to halt itself and move itself from a first processor to another processor that is connected to the first processor for example by a communications network. The computer program is referred to as being autonomous because it is able to "decide" where to move and what it will do independently of external requests. An example of a mobile autonomous software agent is a Java mobile agent. Details about Java mobile agents are given in the article, "Under the Hood: The architecture of aglets", by Bill Venners, JavaWorld April 1997 the contents of which are incorporated herein by reference.

By extending the SIP protocol increased functionality is provided. SIP messages are modified to carry computer software code such as Java applets or to carry an address such as an universal resource locator (URL) indicating where computer software code is stored. An application programming interface (API) is also defined which allows the computer software code to interact with a receiving host system. SIP clients are also modified in order that they execute the computer software code associated with the SIP messages before any other actions are taken as a result of receipt of the SIP message.

FIG. **12** shows a communications network **1001** comprising a plurality of communications network nodes **1010** each such node comprising:

a SIP client **1011**;

an input **1012** arranged to receive SIP messages which may be associated with computer software code; and

a processor **1013** arranged such that in use, when a SIP message is received, any computer software code associated with that SIP message is executed by the processor. This processor is provided by the host system and may comprise a Java virtual machine or any other suitable processor. These communications network nodes are referred to as enhanced SIP nodes because they are arranged to allow the enhanced SIP process to work.

The communications network of FIG. **12** is used in conjunction with the method illustrated in FIG. **13** in order to implement the enhanced SIP process. FIG. **13** is a flow diagram of a method of communicating between a first and a second node in a communications network, each of said nodes comprising a SIP client, said method comprising the steps of:

associating computer software code with a SIP message (box **1020** in FIG. **13**);

sending the SIP message from the first SIP client associated with the first node to the second SIP client associated with the second node (box **1021** in FIG. **13**); and

executing the computer software using the second node (box **1022** in FIG. **13**).

For example, FIG. **14** illustrates an example of how a plurality of enhanced SIP clients **1030**, **1031**, **1032**, **1033**, **1041** interact. Each SIP client is supported on a communications network node (not shown). SIP client A **1030** is connected to SIP client B **1031** via a communications link **1034** and SIP client B **1031** is connected to both SIP client C **1032** and SIP client D **1033** via communications links **1034**. SIP client B **1031** has a host system **1035** which comprises a Java virtual machine. SIP client D **1033** is also connected to SIP client E via a communications link. SIP client D and has a host system **1039** which comprises a Java mobile agent virtual machine and SIP client E **1041** also has a host system **1041** which comprises a Java mobile agent virtual machine **1042**.

Using the enhanced SIP protocol, computer software code such as Java applets are associated with a SIP message **1036**. That is, the computer software code may be added to the SIP message body itself or may be stored separately and an address of the storage location added to the SIP message. It is not essential to use Java applets or Java mobile agents; any other suitable computer software code may be used. The message **1036** is sent from SIP client A **1030** to SIP client B **1031**. SIP client B detects the presence of the Java applets (or other computer software code) associated with the SIP message **1036** and executes these Java applets using its Java virtual machine **1035** (or other type of host processor).

Any suitable method of detecting the presence of computer software code associated with the SIP message **1036** may be used. For example, an indicator may be placed in the header of the SIP message **1036** and the SIP client **1031** arranged to detect that indicator and associate it with the presence of computer software code. An example of such an indicator in a SIP message is described in more detail below.

By executing the Java applets, two new SIP messages **1037**, **1038** are created one of which **1037** contains a Java mobile agent and the other which does not. This is just one example of a something that the computer software code associated with the SIP message could do. For example, the computer software code could also be arranged to modify existing SIP messages, delete existing SIP messages, generate SIP messages, receive SIP messages or to control the SIP client and/or the host processor to perform any other suitable function. The computer software code is arranged to interact with the host processor via an API as described below. Security restrictions may be enforced by the SIP client and or host system in order to limit the actions that any software code associated with a SIP message is able to effect. More detail about these security restrictions is given below.

The executed Java applets then cause SIP client B **1031** to send one of the created messages **1037** to SIP client D **1033** and the other **1038** to SIP client C **1032**. The message **1037** sent to SIP client D contains a Java mobile agent (or other computer software code or an address of computer software code). If SIP client D has the capability to execute the Java mobile agent contained in message **1037** then SIP client D does so. However, if SIP client D does not have this capability, for example, if SIP client D has no Java mobile agent virtual machine, then SIP client D simply follows the standard SIP procedure for unsupported require extensions. This involves returning an error message to SIP client B, indicating that the Java applet in message **1037** was not executed.

In the meantime, SIP message **1038** which is not associated with any computer software code, is sent to SIP client C **1032** and any SIP process associated with that message **1038** is carried out following the standard SIP protocol.

In this example, SIP client D does have an associated Java mobile agent virtual machine **1039** and so when message **1037** arrives, the Java mobile agent in message **1037** begins to execute on this processor. At some point in the execution, the Java mobile agent suspends itself and includes itself in SIP message **1040** which is sent to SIP client E. This is one example of a process that may occur by incorporating a Java mobile agent into a SIP message.

In the enhanced SIP protocol described herein, standard SIP messages are modified by associating computer software code with them as described above. For example, one or more Java applets or Java mobile agents are stored in a multipart MIME section in the body of a SIP message or a URL indicating where the Java applets or Java mobile agents are stored is added to the SIP message.

In some examples, an indicator is added to the SIP message header, in order to indicate that computer software code is associated with that SIP message. For example, a "Require request-header" is used to indicate that Java enhanced SIP must be supported to process a SIP message that is associated with Java applets or Java byte code. This require request header is the same as the header for a standard SIP message except that the content type field in the entity header is used to indicate that the content type is a Java applet or the URL of a Java applet which must be retrieved. Also, the require field of the request-header is used to specify that Java enhanced SIP must be supported to process the message concerned.

FIG. **15** illustrates the structure of a standard SIP message and shows how this structure is used in the improved SIP protocol described herein. The structure of a standard SIP message is illustrated at **1040** in FIG. **15**. Thus a standard SIP message comprises a general-header, a request-header, an entity header, a CRLF and a message body. The structure of a general-header is shown at **1041** in FIG. **15** and similarly the structures of each of an entity header **1042**, request header **1043** and response header **1044** are shown. In order to indicate that the improved SIP protocol described herein is being used markers or tags are included in the SIP message in any suitable location. For example, the content-type field of an entity header may be used to indicate that the content type is a Java applet or the URL of a location of a Java applet. Similarly, the content-type field of an entity header may be used to indicate that the content type is a Java mobile agent or the URL of a location of a Java mobile agent. Also, the require field of a request header may be used to indicate that Java enhanced SIP must be supported to process the message concerned. However, it is not essential to use the content-type field or the require field for this purpose. Any other suitable field(s) may be used.

FIG. **16** shows an example of an INVITE message according to the improved SIP protocol described herein. The content type field contains the words "multipart/mixed" which indicates that the INVITE message body is in the form of a MIME multipart message which contains one or more Java applets or Java mobile agents. The require field contains the words "org.ieff.sip.java-enhanced-sip" which indicate that the improved SIP protocol must be used to process this message. Part of the body of the INVITE message containing the Java applet(s) or Java mobile agents is shown **1050**.

The SIP clients used to implement the improved SIP protocol are the same as standard SIP clients except that they are arranged to do the following things:

Detect improved SIP messages which are associated with computer software code. For example, this may be done by arranging the SIP client to recognise the

presence of the words "org.ietf.sip.java-enhanced-sip" or "org.ieff.sip.java-mobile-agent-enhanced-sip" in the SIP message header.

If an improved SIP message is received and detected, the software code associated with that SIP message is accessed by the SIP client and executed on the SIP client's host processor. Preferably, this execution is carried out immediately, before processing the SIP message any further. For example, if a content type field in a SIP header indicates that a URL for a Java applet is present then the SIP client must immediately get the applet from the URL and execute the applet on a Java virtual machine associated with the SIP client. If the SIP client does not execute the software code then it is preferably arranged to respond by returning status code **420** (bad extension) and by listing org.ieff.sip-.java-enhanced-sip in an unsupported header. The SIP client may not execute the software code if it is unable to do so, for example, if no Java virtual machine is available, or if the SIP client decides not to do this, for example, for security reasons.

Match incoming SIP messages to patterns and in the event of a match "wake up" any waiting computer software code. This is described in more detail below.

The SIP client's host processor is modified as compared to a standard SIP client's host processor in that it must comprise a processor of a specific type. For example, a Java virtual machine in the case that Java applets are associated with the improved SIP messages. In the case that Java mobile agents are used, a Java mobile agent virtual machine is required. Also, the SIP client's host processor has access to or comprises an API to allow the computer software code associated with the improved SIP messages to interact with the SIP client. For example, in the case that Java applets are used, the SIP client's host has access to a set of Java classes or applets that are defined in a Java enhanced SIP API. This API allows access into the SIP client to allow SIP messages to be built and sent subject to security restrictions. Using the API received Java applets or Java mobile agents are able to generate and receive SIP messages using the receiving SIP client.

Passing of control between the computer software code associated with improved SIP messages and the SIP client concerned.

In the case that standard SIP messages are used, these are processed by SIP clients in the standard way and control remains with the SIP clients. However, in the improved SIP case described herein, any computer software code associated with a SIP message takes precedence over other standard SIP processes associated with the SIP message or with any other SIP messages received by a SIP client during processing of the computer software code.

For example, the computer software code associated with a SIP message can be arranged to initiate a SIP session and to wait for a SIP response before proceeding. During this waiting period, control remains with the computer software code. The computer software code is able to specify that it will go to sleep and wait for the next SIP message which matches a particular patter. In that case, the SIP client does no other actions during the sleep period. Alternatively, the computer software code can deal with any other incoming SIP messages itself during the sleep period. Thus control does not pass back to the SIP client until the computer software code wants it to even if SIP messages from other sessions are arriving.

Application Programming Interface (API)

As described above an API is specified in order that the computer software code associated with improved SIP messages is able to affect the SIP client. For example, this API allows a received Java applet or Java mobile agent access to the SIP messaging functions on the SIP client.

Examples of methods that the API supports comprise:

SendSIPMessage—sends a SIP message and establishes a context for the Session if one does not already exist. The invoker (which is the piece of software code which called this function) can indicate if it wants the message to be part of an existing Session. For example, the invoker could be a Java applet or Java mobile agent.

ReceiveSIPMessage—retrieves a SIP message from the Client's input buffer on a first in first out (FIFO) basis.

ReceivedMessageSummary—returns a summary of any received messages in the client's input buffer along with a count of messages received. If the client does not support buffering of input messages this is indicated.

QueryCapabilities—returns the capabilities of the Client. These include the ability to buffer incoming messages and the buffer size.

Querystatus—returns the status of any sessions the client is currently involved in.

MatchMessageAndWake—checks incoming messages against a particular pattern and if they match wakes up the indicated applet or Java mobile agent and passes the messages directly to the indicated applet.

ProcessMessage—sends a message to the Client and passes control to the client for the message to be processed as in standard SIP. For example, this can be used after an applet or Java mobile agent has looked at the message or altered it in some way and then wants to pass the message back to the client to be processed as in standard SIP.

ProcessMessageAndReturn—as for ProcessMessage except that control is passed back to the invoker after the message has been processed.

ProcessFromBufferAndReturn—processes the next message on the INPUT buffer as in standard SIP within the client and then returns control to the invoking applet or Java mobile agent.

Changes to SIP Proxy and SIP Server Behaviour

Following standard SIP as defined in "Request for comments (RFC) 2543 SIP: Session Initiation Protocol", SIP proxy and redirect servers must ignore features that are not understood. That is, if a SIP proxy or redirect server is not arranged to understand the improved SIP messages described herein then it must ignore features of those messages that are not common to standard SIP. A SIP proxy server is a communications network node which communicates using the SIP protocol on behalf of other parties. A SIP redirect server is a communications network node which receives SIP messages and directs these to another communications network node. If a particular extension to the standard SIP protocol requires that intermediate devices support it, the fact that the extension is being used must be tagged in the proxy-require field as well (see section 6.28 of the SIP RFC mentioned above). Thus for the improved SIP described herein, an indicator is placed in the proxy-require field to specify that the improved SIP is being used.

Security

Preferably, security mechanisms are incorporated in to the improved SIP protocol although this is not essential. For example, a host system which supports a SIP client preferably comprises security mechanisms for controlling the

activity of software code such as Java applets or Java mobile agents received as a result of the improved SIP messages. These security mechanisms may be configured by a user or operator, for example, to always allow or prevent certain operations from being carried out by Java applets or Java mobile agents received from improved SIP messages. The user may datafill a matrix of SIP operations against security mechanism actions. It is also possible for the security mechanism to prompt the user to ask for permission to proceed with certain actions. The security mechanisms are put into effect by a security manager which takes the form of a computer software application located at each SIP client. Preferably, all the methods specified in the API are arranged to check with the security manager at the SIP client concerned before proceeding with the rest of that method. In the case that Java byte code, Java applets or Java mobile agents are used, then the security mechanisms are preferably designed to conform to the standard Java security practices.

An example of an algorithm for a security mechanism is:

Index the matrix for user defined security checks against that operation

Extract the method corresponding to the security action datafilled by the user

Execute that security mechanism method

If the result of the security mechanism method is "pass" then continue and call the SIP API method

Else display a security disallowed message and return without calling the SIP API method.

Actions that a user may datafill for a given SIP operation include:

Allow always

Disallow always

Allow conditional

Disallow conditional

Prompt y/n

Allow and display warning or info

An example of use of the improved SIP protocol to create a service for automatically setting up multimedia conferences is now described.

Conferencing System

Using the improved SIP protocol a conferencing service is created whereby a single chairperson is able to set up the conference by sending out SIP INVITE messages. The method is suitable for multimedia conferences. The INVITE messages are associated with computer software code which executes on the host machines of invited attendees to set up the conference call. This greatly simplifies the process of setting up a conference call such as a multimedia conference call.

For example, the computer software code associated with the improved SIP INVITE messages can be arranged to set up connections from each attendee's machine to several video sources and to an electronic whiteboard to be shared for the meeting. The computer software code can also be arranged to start up a web browser to a page relevant to the meeting on each attendee's machine. As well as this the computer software code is able to set up all the audio paths between all the parties with everyone but the chairman initially on mute. As well as this the computer software code is able to take into account different capabilities of individual attendee's host machines. For example, a particular attendee such as a mobile caller may only have audio capabilities whilst a full multimedia caller may have audio, video, data and web capabilities. In order that these capabilities are taken into account, attendee's indicate what their capabilities are in SIP messages as required.

The multimedia conferencing service is particularly advantageous from the attendee's point of view. All the attendee has to do is to accept the incoming call and SIP INVITE message and everything will be set up for them automatically. Alternatively, the attendee may call a conference number and receive a SIP message in reply which is associated with the required computer software code. The conference number may be the number of a particular user client or of a central conference service provider.

Preferably security mechanisms are used in the multimedia conferencing service as described above.

FIG. **17** is a flow diagram of a method where a central conference service system is used and where Java applets are associated with the improved SIP messages. The first stage involves a user who wants to join a conference call sending a SIP INVITE message to the conference service system from his or her terminal (box **1060** FIG. **17**). This call is received by the conference service system which then returns an acknowledgement message ACK back to the user's terminal (box **1061** FIG. **17**). This ACK message is associated with one or more Java applets which contain methods from the API discussed above. The user's SIP client receives the ACK message, accesses the associated Java applet(s) and runs these using its associated Java virtual machine (box **1062** FIG. **17**).

The Java applet(s) query the exact capabilities of the user's SIP client and host machine and taking these capabilities into account, initiate SIP sessions for any audio, video and data streams associated with the conference as appropriate given the capabilities (box **1063** of FIG. **17**). Depending on how the user has his or her security mechanisms set he or she may be prompted before the sessions are set up for the various media streams. When the Java applet(s) initiate the SIP sessions (box **1063** of FIG. **17**) they may also be arranged to set up these SIP sessions such that all the attendees except for a chairperson are on mute. This is particularly advantageous, because the chairperson is then easily able to announce the beginning of the meeting and to chair the meeting in an organised fashion.

The Java applets(s) may also be arranged to forward details of a web page from each attendee to a chairperson or to the conference service system. For example, a web page giving biographical details of each attendee may be forwarded to a chairperson who then makes these available to each other attendee. In a similar manner, digital photographs of each attendee may be forwarded to the chairperson by the Java applets. It is also possible for the Java applets to request a joining message from each attendee which is then forwarded to a chairperson automatically by the Java applets. This joining message may contain security requirements specific to each attendee.

Depending on the number of parties to the conference, a conferencing bridge facility may be used as is known in the art. Alternatively, a software based technique is used to connect the parties to the conference.

An example of an algorithm that is encoded in the Java applet(s) of the method described immediately above is:

Read the message that the Java applet was associated with to obtain the addresses for the various streams in the call

Query the capabilities of the SIP client

Query the capabilities of the host system

Based on the above information for each media type and application available on the conference call:

If this application and media type is supported on the SIP client, initiate a SIP session between the SIP client and the relevant SIP client for that media stream.

Initiate a SIP message to the central conference service system detailing the number and types of streams set up.

FIG. **18** is a flow diagram of a method of setting up a conference call between two or more parties, each party comprising a SIP client and a host processor, said method comprising the steps of:

associating computer software code with a SIP message (box **1070** of FIG. **18**);

sending the SIP message to each of the parties (box **1071** of FIG. **18**);

executing the computer software code at each of the host processors (box **1072** of FIG. **18**).

FIG. **12** also shows a system for automatically setting up a conference call between two or more parties **1010**, each party comprising a SIP client **1011** and a host processor **1013**, said system comprising:—a processor **1013** for associating computer software code with a SIP message and to send that SIP message to each of the parties **1010**; and wherein each of said host processors **1013** is arranged to execute the computer software code in use, when the SIP message is received.

In the case that a conferencing system is used, this system sends the SIP messages to each party as a result of request calls from those parties to the system. In the case that a chairperson sets up the call, then the chairperson sends the SIP messages to each party.

Hunt Group System

An example of the use of improved SIP with Java mobile agents is now described. In this example, a service is provided whereby an automated system calls several telephones within a defined group (such as a team in an office) until one of those telephones is answered. For example, the nodes of the communications network in FIG. **12** may each provide a telephone implemented by software in the SIP clients **1011**. Each telephone within the group **1001** comprises a SIP client **1011** and a host processor **1013** as illustrated in FIG. **12** and the telephones are connected to one another via a communications network **1001** as shown in FIG. **1012**. The host processors each comprise a Java mobile agent virtual machine.

A user, which may be an automated service or a human using a terminal connected to the communications network **1001**, telephones one of the telephones **1010** within the defined group. If the called telephone is not answered after a specified number of rings or an elapsed time, then software at the SIP client **1011** of the called telephone creates a Java mobile agent, associates this with a SIP message, and sends the SIP message to a predefined second SIP client. This second SIP client is one of the telephones within the defined group **1001**.

The second SIP client receives the SIP message which is associated with the Java mobile agent. The Java mobile agent then executes itself on the Java mobile agent virtual machine associated with the second SIP client. The Java mobile agent is arranged to apply ringing to the second telephone and queries the second telephone's identification details and sends these back to the original caller. If the caller is using a host processor that has a display system associated with it, then information about the call and the fact that it has been forwarded to the second telephone in the defined group is sent by the Java mobile agent to this display.

If the second SIP client does not answer after a specified number of rings or time then the second SIP client repeats the method that the first SIP client carried out as described above. However, the second SIP client incorporates information about the fact that the call has been forwarded again.

After the method has been repeated a pre-determined number of times and if the call is not answered, then the call is sent back to the first SIP client that was called. A display of the route taken and the fact that the call was not answered is made at the first SIP client if a display is available.

If the call is answered, information about the route taken and the identity of the answering SIP client is sent back to the caller which may be an automated service.

FIG. **21** shows a method of forwarding a call from a first SIP client to a second SIP client, each of said SIP clients being associated with a host processor, said method comprising the steps of:

receiving a call at the first SIP client and if that call is not answered then associating computer software code with a SIP message said computer software code being arranged to forward a call (box **10100** FIG. **21**);

sending the SIP message from the first SIP client to a specified second SIP client (box **10101** FIG. **21**); and

executing the computer software using the host processor associated with the second SIP client such that the call is forwarded to the second SIP client (box **10102** FIG. **21**).

Client test system

Another example of the use of Java mobile agents with improved SIP involves a test system for a pre-defined group of SIP clients. For example, the network of SIP clients shown in FIG. **12**. The SIP clients **1011** are connected to one another to form a communications network **1001** as illustrated in FIG. **21**. Each SIP client **1011** is associated with a host processor **1013** which comprises a Java mobile agent virtual machine.

A test system (for example, software located at one of the nodes **1010** in the communications network **1001**), which may be an automated software service, creates a Java mobile agent, associates this with a SIP message, and sends that SIP message to one of the SIP clients **1011** in the group. The Java mobile agent executes on the receiving SIP client and sets up one or more test sessions. The results of these test sessions are stored by the Java mobile agent in its private data, together with any other required information. The Java mobile agent then associates itself with another SIP message and arranges that this SIP message be sent to another SIP client in the group. When the SIP message reaches another SIP client the process of obtaining information is repeated so that more information is added to the Java mobile agent's private data. Another SIP message is used to send the Java mobile agent on to another SIP client and so on, until all the SIP clients in the group have been visited. Once all the SIP client's in the group have been visited by the Java mobile agent, this agent associates itself with a SIP message in order to be sent back to the originating SIP client. In this way the Java mobile agent is able to report the results of its tests to the originating SIP client. The Java mobile agent may also be arranged to initiate other actions to fix any faults that it finds as it finds them. FIG. **20** shows a method of testing members of a group of SIP clients each SIP client being associated with a host processor said method comprising the steps of:

associating computer software code suitable for said testing with a SIP message (box **1090** FIG. **20**);

sending the SIP message one of the SIP clients (box **1091** FIG. **20**);

executing the computer software at the host processor associated with that SIP client in order to obtain test results (box **1092** FIG. **20**); and

repeating steps (ii) to (iii) for each of the other SIP clients in the group (box **1093** FIG. **20**).

Upgrade or replacement of SIP clients

Consider a situation in which it is required to upgrade or replace SIP clients which support the improved version of SIP described herein. This may be carried out automatically as follows:

The software for the upgrade or new SIP client is associated with a SIP message, for example, by building the software into a Java applet and adding this applet to a SIP message. This SIP message is then sent to all the SIP clients which are to be upgraded or replaced. On receipt of the SIP message at a SIP client, the existing SIP client runs the software code in order to effect the upgrade or replacement. The extent to which the upgrade or replacement is effected depends on the security specifications and the type of SIP client. By using the improved SIP protocol in this way, upgrades or replacement of a plurality of SIP clients is achieved quickly and easily.

FIG. **19** shows a method of upgrading or replacing interconnected SIP clients each SIP client being associated with a host processor said method comprising the steps of:—

associating computer software code suitable for said upgrade or replacement with a SIP message (box **1080** FIG. **19**);

sending the SIP message to each of the SIP clients (box **1081** FIG. **19**); and

executing the computer software at each of the host processors (box **1082** FIG. **19**).

What is claimed is:

1. A method of remotely controlling a destination terminal from an originating terminal said destination terminal having an associated signalling protocol client and an associated processor comprising the steps of:

(i) storing computer software code in at least one signalling protocol message;

(ii) sending the signalling protocol message to the destination terminal from the originating terminal;

(iii) executing the computer software code using the processor associated with the destination terminal in order that the originating terminal controls the destination terminal.

2. A method as claimed in claim **1** wherein said step (iii) of executing further comprises activating a security means at the destination terminal and executing the computer software code depending on the activated security means.

3. A method as claimed in claim **1** wherein said computer software code is arranged to access information about the identity of a caller.

4. A method as claimed in claim **3** wherein said computer software code is further arranged to display the identity information at the destination terminal.

5. A method as claimed in claim **1** wherein said computer software code is arranged to access information about a priority level for a call associated with the signalling protocol message.

6. A method as claimed in claim **1** wherein said computer software code is arranged to detect whether the destination terminal is engaged, and if so to clear the destination terminal in order that it is able to accept an incoming call associated with the signalling protocol message.

7. A method as claimed in claim **1** wherein said computer software code is arranged to access information from the destination terminal about the configuration of that terminal.

8. A method as claimed in claim **7** wherein said computer software code is further arranged to control the destination terminal on the basis of accessed configuration information.

9. A method as claimed in claim **1** wherein said computer software code is arranged to modify the configuration of terminating services associated with the destination terminal.

10. A method as claimed in claim **1** wherein said computer software code is arranged to direct a call associated with the signalling protocol message to a voice mail system associated with a called party.

11. A method as claimed in claim **1** wherein said signalling protocol message is a session initiation protocol (SIP) message and wherein said computer software code is selected from: Java byte code, Java applets and mobile automated software agents.

12. A destination terminal comprising:—

(i) a signalling protocol client arranged to receive one or more signalling protocol messages sent from an originating terminal;

(ii) a processor arranged to access any computer software code stored in received signalling protocol messages in use; and wherein said processor is arranged to execute such accessed computer software code such that the destination terminal is controlled.

13. A destination terminal as claimed in claim **12** which further comprises stored security information and wherein said processor is arranged to check said security information before executing the accessed computer software code.

* * * * *

EXHIBIT D

US006937572B1

(12) **United States Patent**
Egan et al.

(10) **Patent No.: US 6,937,572 B1**
(45) **Date of Patent: Aug. 30, 2005**

(54) **CALL TRACE ON A PACKET SWITCHED NETWORK**

(75) Inventors: **Brian B. Egan**, Ontario (CA); **Milos Vodsedalek**, Ontario (CA)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 794 days.

(21) Appl. No.: **09/751,796**

(22) Filed: **Dec. 29, 2000**

(51) **Int. Cl.**[7] **G01R 31/08**
(52) **U.S. Cl.** **370/252**; 709/224
(58) **Field of Search** 370/252–254, 370/260–9, 351–7, 469; 709/223–5; 345/732–745

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,144,336 A * 11/2000 Preston et al. 342/357.09
6,636,504 B1 * 10/2003 Albers et al. 370/352
6,643,267 B1 * 11/2003 Karia et al. 370/252
6,681,232 B1 * 1/2004 Sistanizadeh et al. 707/104.1
6,721,184 B1 * 4/2004 Lehmeier et al. 361/728
2002/0114274 A1 * 8/2002 Sturges et al. 370/229

OTHER PUBLICATIONS

Stevens, Richard; TCP/IP Illustrated, vol. 1; 1994; Addison-Wesley, pp. 263-265, 97-109.*

* cited by examiner

Primary Examiner—David Vincent
(74) *Attorney, Agent, or Firm*—Mintz Levin Cohn Ferris Glovsky & Popeo P.C.

(57) **ABSTRACT**

A call trace feature for a network compatible device, such as a voice over Internet Protocol device. The call trace information may include an Internet Protocol (IP) address, a geographical location of the end-point device, a type or class of the end-point device, a call route, a topology of the route, a domain name server of the IP address and route, a directory number and name, a call back number, an advisement as to whether the IP address for the end-point device is mobile and an advisement as to what redirection may have occurred before the call was completed.

26 Claims, 1 Drawing Sheet

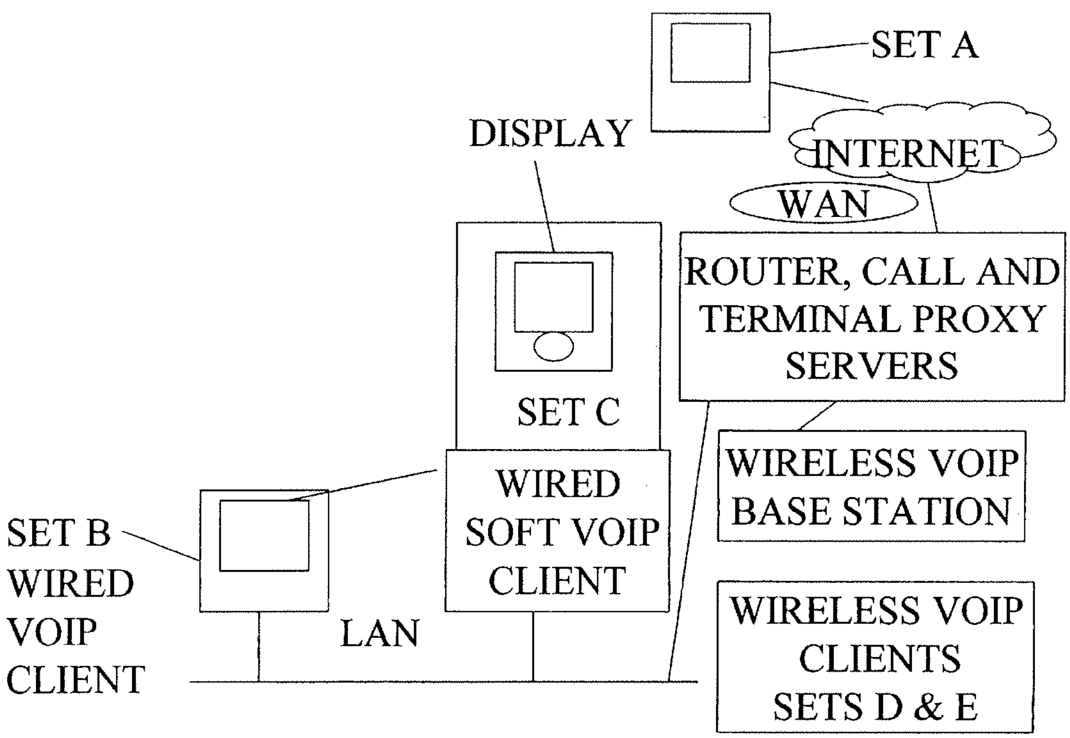
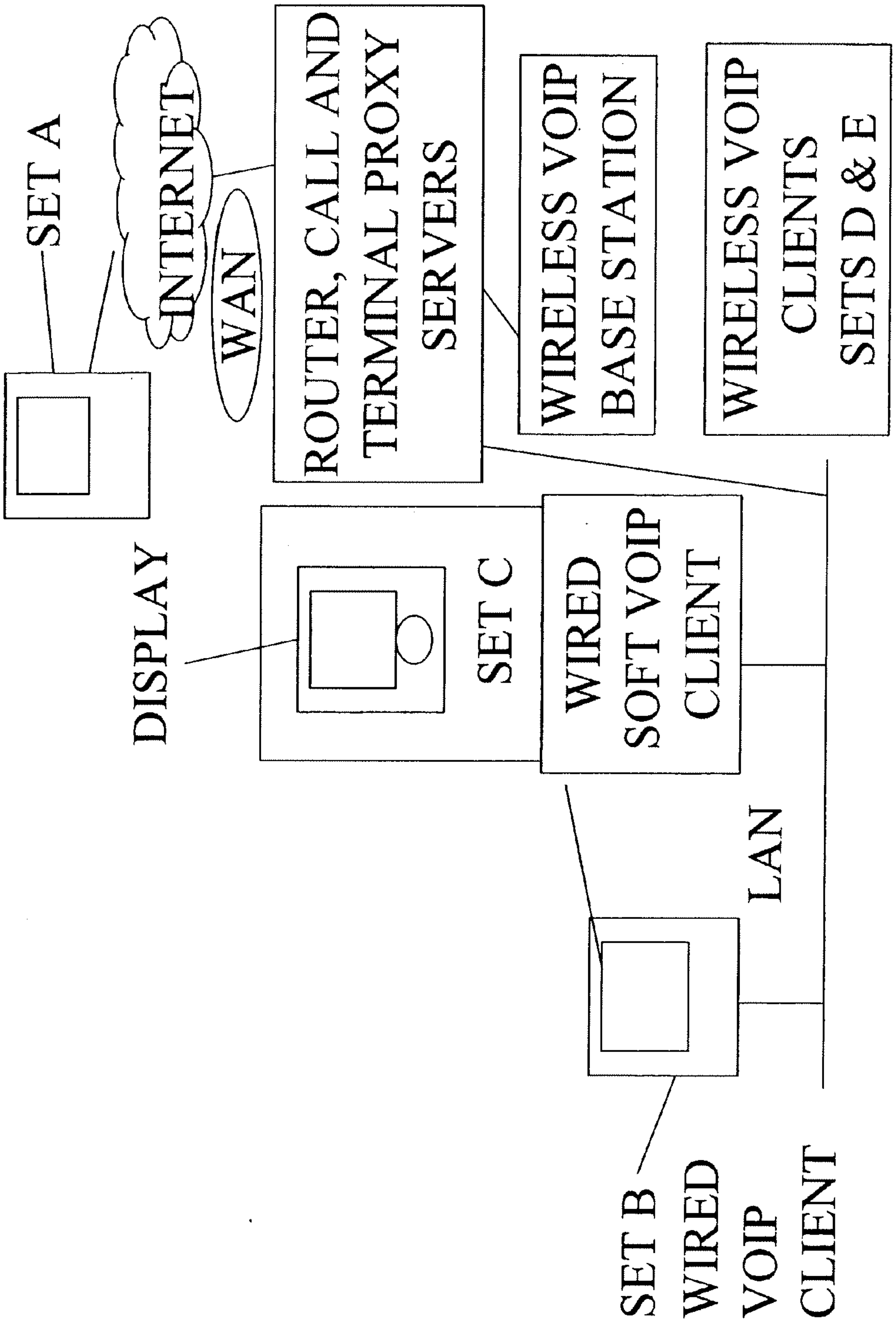


FIG. 1


SET A
INTERNET
WAN
ROUTER, CALL AND TERMINAL PROXY SERVERS
WIRELESS VOIP BASE STATION
WIRELESS VOIP CLIENTS SETS D & E
DISPLAY
SET C
WIRED SOFT VOIP CLIENT
LAN
SET B WIRED VOIP CLIENT

CALL TRACE ON A PACKET SWITCHED NETWORK

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates to providing call trace information on a packet switched network to users of various network compatible devices, such as telephone sets, personal digital assistants, soft phones and the like.

2. Discussion of Related Art

Support for end-to-end voice calls using the Internet as an alternative to traditional public switched telephone networks (PSTN) is well known. Unlike the PSTN, which is circuit-switched, the Internet is packet-switched; communication on the Internet is accomplished by transmitting and receiving packets of data.

In addition to data, each packet contains an address to ensure that it is routed correctly. The format of these packets is defined by the Internet Protocol (IP).

One type of allowable data is encoded, digitized voice. Voice over IP (VoIP) is encoded digitized voice that is packetized in accordance with IP, and communicated over the Internet for telephone-like communication.

A service provider can off-load branch-office voice traffic from the PSTN and route it across the company's existing packet switched data intranet, thereby eliminating toll charges. In addition, interoffice fax traffic can be routed across a company's data network or through an extranet, using existing fax machines, key systems, Centrexes, and PBXs, making the process of placing calls over the IP network transparent to users.

Users of VoIP devices currently are not provided with call trace information. Such call trace information could be useful in a variety of applications.

For instance, it would be useful for emergency E911 (electronic 911) service to help emergency personnel determine critical information about an IP client in advance of rendering emergency service. Such call trace information would also be useful for tracking down and apprehending prank callers who use VoIP. Even misbehaving IP clients could be identified and reprimanded or shut down by network administrators that have access to such call trace information. In addition, call trace information may serve as critical debugging information to help network servicing personnel determine network faults and the like.

The name and number information of the called party may be linked to address and geographic location information about a called party. If geographic information about the called party of a VoIP call were known to the caller, the caller would then be in a position to know whether the call is costly due to the geographic distance from the called party and whether the call to be placed is being received in a different time zone. If the caller is a retail merchant, geographical information about the called party may be useful in deciding whether to ship certain products to certain geographical zones.

It would be advantageous to provide call trace information to users and/or administrators of network compatible devices. Such users and/or administrators would then be informed about a caller's end-point device in advance of taking some action with respect to that caller or caller's end-point device.

SUMMARY OF THE INVENTION

An aspect of the invention relates to providing call trace information about a caller's end-point device that communicates on a packet switched network. The end-point device may be a voice over Internet Protocol (VoIP) device, which operates using voice that is packetized in accordance with Internet Protocol (IP) and transmitted over the Internet.

The call trace information may include the following about the end-point VoIP device: the IP address, the geographical location of the end-point device and the type or class of the end-point device. In addition, the call trace information may provide the call route and the topology of the route, the domain name server (DNS) of the IP address and route, the directory number and name and the call back number for the user. Further, the call trace information may inform about whether the IP address for the end-point device is mobile and what redirection may have occurred before the call was completed.

In the case of a conference call, a further aspect of the invention is to provide a feature that enables call trace information about each one of the conferees to be delivered back to the originator of the conference call, whether the conference is a voice, video or chat session. Even a circuit switched time division multiplex (TDM) device may activate this feature if it is connected to an end-point VoIP device through a gateway.

BRIEF DESCRIPTION OF THE DRAWING

For a better understanding of the present invention, reference is made to the following description and accompanying drawings, while the scope of the invention is set forth in the appended claims.

The drawing is a schematic representation of a network having network compatible devices.

DETAILED DESCRIPTION OF THE INVENTION

While the following description makes reference to providing call trace information to compatible VoIP devices, the call trace information may be provided to any packet switched network device.

The drawing illustrates a communication network having five VoIP clients. The VoIP clients each include network compatible devices, such as voice communication devices, referred to as sets A, B, C, D & E. The sets may receive voice communications over a packet switched network, such as the Internet.

The configuration depicted in the drawing is illustrative only; any number of communication devices may be employed and different types of networks may be configured instead or in addition to that illustrated. For instance, the user may have a plain old telephone service (POTS) telephone with an adjunct device that connects to a VoIP gateway that is in communication with the Internet or some other IP network.

In the configuration depicted in the drawing, set A talks to set B through a local area network (LAN) and a wide area network (WAN). A call server and a terminal proxy server control sets A, B, C, D & E. Sets C, D & E may be in conference with sets A & B or have separate calls going on at the same time.

Sets D & E, which are wireless VoIP clients, communicate through a wireless VoIP base station with the Sets A, B or C. Set C may be a computer telephony integration with a

monitor that displays a VoIP control panel. Set D may be a wireless LAN (802.11) Voice over IP phone with an integral text display capability. Set E may be a third generation wireless digital phone with an integral text and video display capability.

Where the adjunct device for a POTS telephone is being used instead of a VoIP device, the adjunct device would be capable of handling the same type of information that the VoIP device handles.

A user may have a VoIP device or a TDM device (such as one connected through a gateway to a packet switched network). The user of the packet switched network activates the call trace information feature of the present invention. The activation may be by depressing a button, sliding a switch, or entering a feature code. Activation may also be automatic—either at startup, during the entire call, or periodically. If desired, the call trace feature may be remotely activated with a sniffer software that is turned on in the network.

Once activated, call trace information is provided at the user's VoIP or TDM device. If desired, the call trace feature may be configured to present only selected pieces of information. The presentation of the information may be in graphic or numeric, textual or any combination thereof. Where the sniffer software is turned on, the sniffer software monitors and reports this information to security or other authorized recipients.

The call trace information may include the following information about the connected end-point; the IP address, the geographical location of the end-point device and the type or class of the end-point device. In addition, the call trace information may provide the call route and the topology of the route, the domain name server (DNS) of the IP address and route, the directory number and name and the call back number for the user. Further, the call trace information may include whether the IP address for the end-point device is mobile and what redirection may have occurred before the call was completed.

In the case of a conference call, the invention may provide: a feature that enables the call trace information for each of the conferees to be delivered back to the originator of a conference call. Even a circuit switched time division multiplex (TDM) device may activate this feature if it is connected to a VoIP gateway.

An intelligent call log capability may be activated to log all of this call trace information for security or network performance analysis. This capability may be triggered at the time the request for call trace information is received by or acted upon by the terminal proxy server or at the time of receipt of the call trace information by the requesting device.

The implementation of this call trace feature uses ping and net trace tools and makes queries to an Internet Protocol based Private Branch Exchange (IP-PBX) call manager or gateway for information such as directory number (and name) to IP translation and for other parameters such as geographical location, time, etc. The IP translation is a gatekeeper function to provide translation between an IP address and a Name or Directory Number. It serves the same type of function as a Domain Name Server (DNS).

A ping tool includes a routine that sends a packet onto the network and obtains a value of the average delay encountered by that packet in reaching the destination and returning. It also confirms the status of the IP end-point.

A network trace tool includes a routine that captures and records events and states that occur while the network is

operating. Events in a trace chronology may be paired, forming event-pairs. An example of an event-pair includes the start and end times.

Implementing the user interface to obtain this call trace information from the IP-PBX call manager or gateway is performed in the same manner implementing any other feature code activated IP-PBX feature.

Most of the call trace information is configured at the time of registration for the VoIP device and stored in a variety of locations or centralized directory and may be accessed using Local Directory Application Protocol (LDAP). As a VoIP device registers with a packet switched network, it is assigned an IP address, a domain name server (DMS) entry, a directory number (DN), a name, and optionally a geographic location that is assigned for static terminals (i.e., those which are at fixed locations and not mobile), and terminal type (e.g., wired, PC based, wireless, etc.). DN, name and terminal type are stored in a call server data base (PBX), while data information such as IP address, DNS entry and other data parameters are stored in a data network server, such as a WINDOWS NT server. The terminal proxy server may retrieve all this information at initial registration of the VoIP device or the terminal proxy server may have dynamic access to this information.

Once the user interface on the VoIP device has been activated, a request signal is sent to the terminal proxy server for call trace information pertaining to an end-point device. In response to the request signal, the terminal proxy server checks its own data base to retrieve the information pertaining to call trace or dynamically retrieves it from the data bases that represent administered data for that end-point device.

The terminal proxy server may supplement the retrieved information by activating a dynamic IP trace against the requested end-point device. The retrieved information is then returned to the VoIP device that requested the call trace information using an agreed upon protocol.

While the foregoing description and drawings represent the preferred embodiments of the present invention, it will be understood that various changes and modifications may be made without departing from the spirit and scope of the present invention.

What is claimed is:

1. Apparatus that obtains call trace information, comprising:

a network compatible device that is configured to communicate over a packet switched network with an end-point device, the network compatible device being configured to generate a request for call trace information that pertains to the end-point device and to receive the call trace information that was requested and to dynamically display at least a portion of the call trace information that was received.

2. An apparatus as in claim 1, wherein the call trace information is selected from a group consisting of an Internet Protocol (IP) address, a geographical location of the end-point device, a type or class of the end-point device, a call route, a topology of the route, a domain name server of the IP address and route, a directory number and name, a call back number, an advisement as to whether the IP address for the end-point device is mobile and an advisement as to what redirection may have occurred before the call was completed.

3. An apparatus as in claim 1, wherein the network compatible device is configured to originate a conference call with a plurality of end points and to receive the call trace information for each of the plurality of end-points.

4. An apparatus as in claim 3, wherein the network compatible device is a circuit switched time division multiplex (TDM) compatible device.

5. An apparatus as in claim 1, wherein the network compatible device is a circuit switched time division multiplex (TDM) compatible device.

6. An apparatus as in claim 1, wherein the network compatible device is a voice over Internet Protocol compatible device.

7. An apparatus as in claim 1, further comprising at least one data base containing the call trace information.

8. An apparatus as in claim 2, further comprising a call log that logs all the call trace information.

9. An apparatus that obtains call trace information, comprising:

a network compatible device that is configured to communicate over a packet switched network with an end-point device, the network compatible device including means for generating a request for call trace information about the end-point device, means for receiving the call trace information that was requested and means for dynamically displaying at least a portion of the call trace information that was received.

10. An apparatus as in claim 9, wherein the call trace information is selected from a group consisting of an Internet Protocol (IP) address, a geographical location of the end-point device, a type or class of the end-point device, a call route, a topology of the route, a domain name server of the IP address and route, a directory number and name, a call back number, an advisement as to whether the IP address for the end-point device is mobile and an advisement as to what redirection may have occurred before the call was completed.

11. An apparatus as in claim 9, wherein the network compatible device is configured to originate a conference call with a plurality of endpoints and to receive the call trace information for each of the plurality of end-points.

12. An apparatus as in claim 10, wherein the network compatible device is a circuit switched time division multiplex (TDM) compatible device.

13. An apparatus as in claim 9, wherein the network compatible device is a circuit switched time division multiplex (TDM) compatible device.

14. An apparatus as in claim 9, wherein the network compatible device is a voice over Internet Protocol compatible device.

15. An apparatus as in claim 9, further comprising means for storing the call trace information.

16. An apparatus as in claim 9, further comprising means for logging the call trace information.

17. A method that obtains call trace information, comprising:

communicating over a packet switched network between a network compatible device and an end-point device, generating a request for call trace information that pertains to the end-point device, subsequently receiving the call trace information that was requested, and dynamically displaying at least a portion of the call trace information that was received.

18. An apparatus as in claim 17, wherein the call trace information is selected from a group consisting of an Internet Protocol (IP) address, a geographical location of the end-point device, a type or class of the end-point device, a call route, a topology of the route, a domain name server of the IP address and route, a directory number and name, a call back number, an advisement as to whether the IP address for the end-point device is mobile and an advisement as to what redirection may have occurred before the call was completed.

19. A method as in claim 18, further comprising logging the call trace information.

20. A method as in claim 18, further comprising storing the call trace information.

21. A method as in claim 18, further comprising originating a conference call with a plurality of end-points and to receive the call trace information for each of the plurality of end-points.

22. A method as in claim 21, wherein the network compatible device is a circuit switched time division multiplex (TDM) compatible device that accesses the packet switched network through a gateway.

23. A method as in claim 18, wherein the network compatible device is a circuit switched time division multiplex (TDM) compatible device that accesses the packet switched network through a gateway.

24. A method as in claim 18, wherein the network compatible device is a voice over Internet Protocol compatible device.

25. An apparatus as in claim 1, further comprising:

a terminal proxy server, comprising software responsive to a request to download call trace information and to transmit the downloaded call trace information to a network compatible device, the call trace information selected from a group consisting of an Internet Protocol (IP) address, a geographical location of the end-point device, a type or class of the end-point device, a call route, a topology of the route, a domain name server of the IP address and route, a directory number and name, a call back number, an advisement as to whether the IP address for the end-point device is mobile and an advisement as to what redirection may have occurred before the call was completed.

26. An apparatus as in claim 1, further comprising:

a terminal proxy server, comprising software responsive to a request for call trace information to dynamically access and then transmit the call trace information to a network compatible device, the call trace information selected from a group consisting of an Internet Protocol (IP) address, a geographical location of the end-point device, a type or class of the end-point device, a call route, a topology of the route, a domain name server of the IP address and route, a directory number and name, a call back number, an advisement as to whether the IP address for the end-point device is mobile and an advisement as to what redirection may have occurred before the call was completed.

* * * * *

EXHIBIT E

US007177399B2

(12) **United States Patent**
Dawson et al.

(10) **Patent No.: US 7,177,399 B2**
(45) **Date of Patent: Feb. 13, 2007**

(54) **DETERMINING THE GEOGRAPHICAL LOCATION FROM WHICH AN EMERGENCY CALL ORIGINATES IN A PACKET-BASED COMMUNICATIONS NETWORK**

(75) Inventors: **Martin C Dawson**, West Wollongong (AU); **Mark Lewis**, Aurora, IL (US); **Maciej Broda**, Ottawa (CA)

(73) Assignee: **Nortel Network Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/861,194**

(22) Filed: **Jun. 4, 2004**

(65) **Prior Publication Data**

US 2005/0190892 A1 Sep. 1, 2005

Related U.S. Application Data

(60) Provisional application No. 60/548,746, filed on Feb. 27, 2004.

(51) **Int. Cl.**
H04M 11/00 (2006.01)

(52) **U.S. Cl.** **379/45**; 370/352

(58) **Field of Classification Search** 379/37–51, 379/90.01; 370/259, 351, 352
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,587,545 B1 7/2003 Antonucci et al.
6,678,357 B2 * 1/2004 Stumer et al. 379/45
6,707,888 B1 * 3/2004 Cope 379/45
6,744,856 B2 * 6/2004 Karnik et al. 379/45
6,940,950 B2 * 9/2005 Dickinson et al. 379/45
2003/0086539 A1 * 5/2003 McCalmont et al. 379/45
2003/0108175 A1 6/2003 Poikselka et al.
2003/0227922 A1 12/2003 Horvath et al.
2005/0063519 A1 * 3/2005 James 379/45
2005/0169248 A1 * 8/2005 Truesdale et al. 370/352

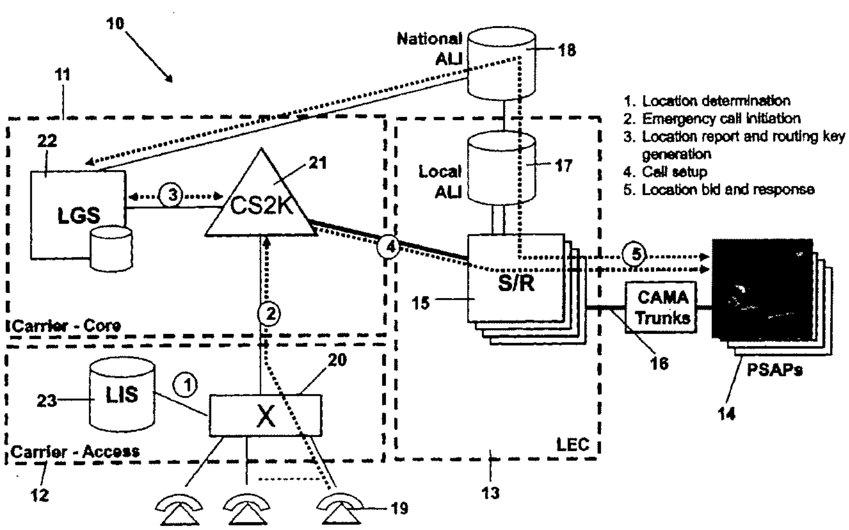
* cited by examiner

Primary Examiner—Stella Woo
(74) *Attorney, Agent, or Firm*—Barnes & Thornburg LLP

(57) **ABSTRACT**

In order that emergency service vehicles can be dispatched to the correct destination promptly, accurate information about the location of the caller is needed. Another problem concerns routing emergency calls to the correct destination. For emergency calls a universal code is used such as 911 in North America and 112 in Europe. This universal code cannot be used to identify the destination of the call. These problems are particularly acute for nomadic communications systems such as voice over internet protocol communications networks. That is because user terminals change network location. These problems are solved by enabling the geographical location of the emergency caller to be determined by entities within a packet-based network without the need for modification of existing emergency services network infrastructure.

10 Claims, 8 Drawing Sheets



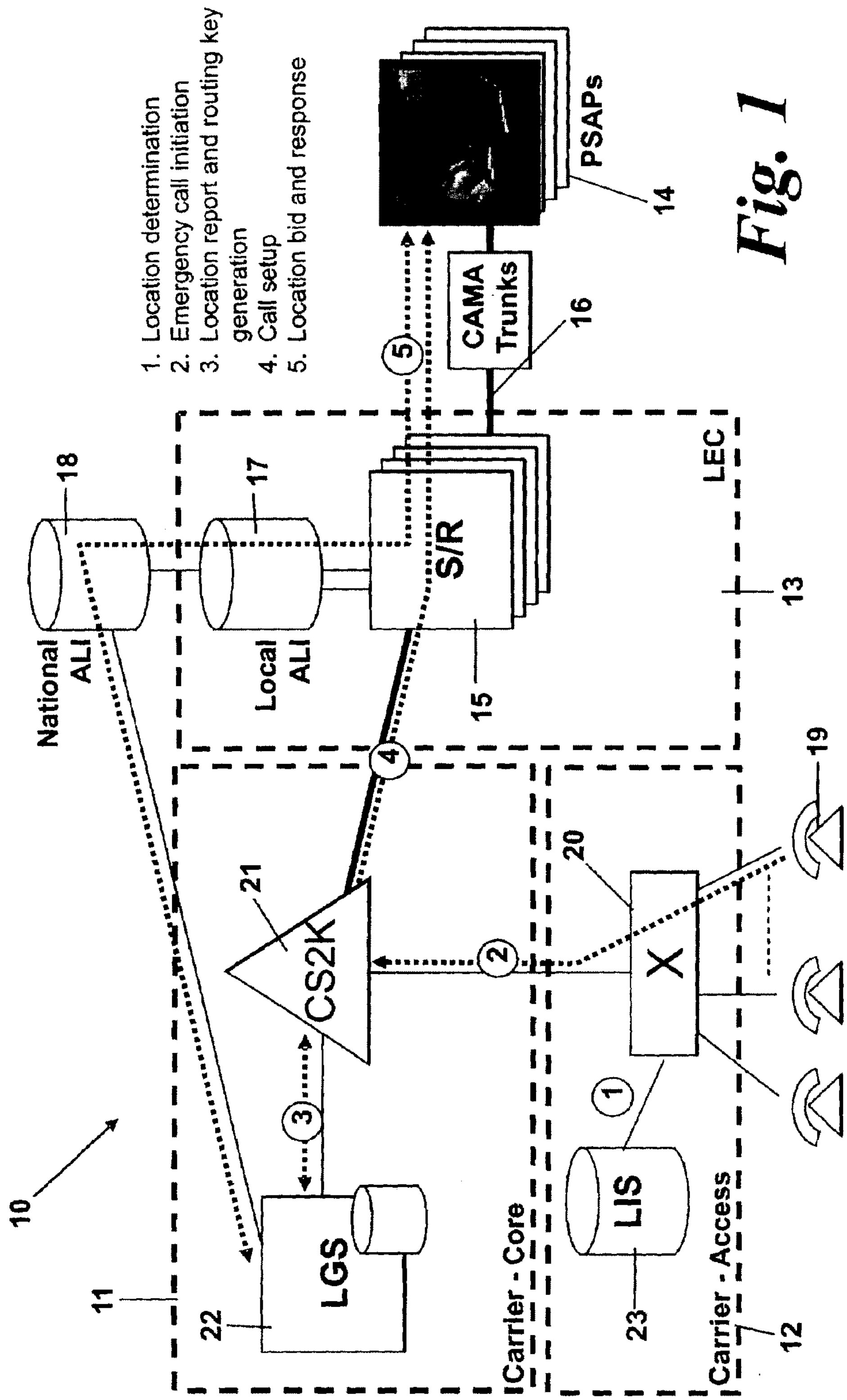

1. Location determination
2. Emergency call initiation
3. Location report and routing key generation
4. Call setup
5. Location bid and response
National ALI
18
Local ALI
17
S/R
15
LEC
13
CAMA Trunks
16
PSAPs
14
10
11
LGS
22
CS2K
21
X
20
19
LIS
23
Carrier - Core
Carrier - Access
12
1
2
3
4
5

Fig. 1

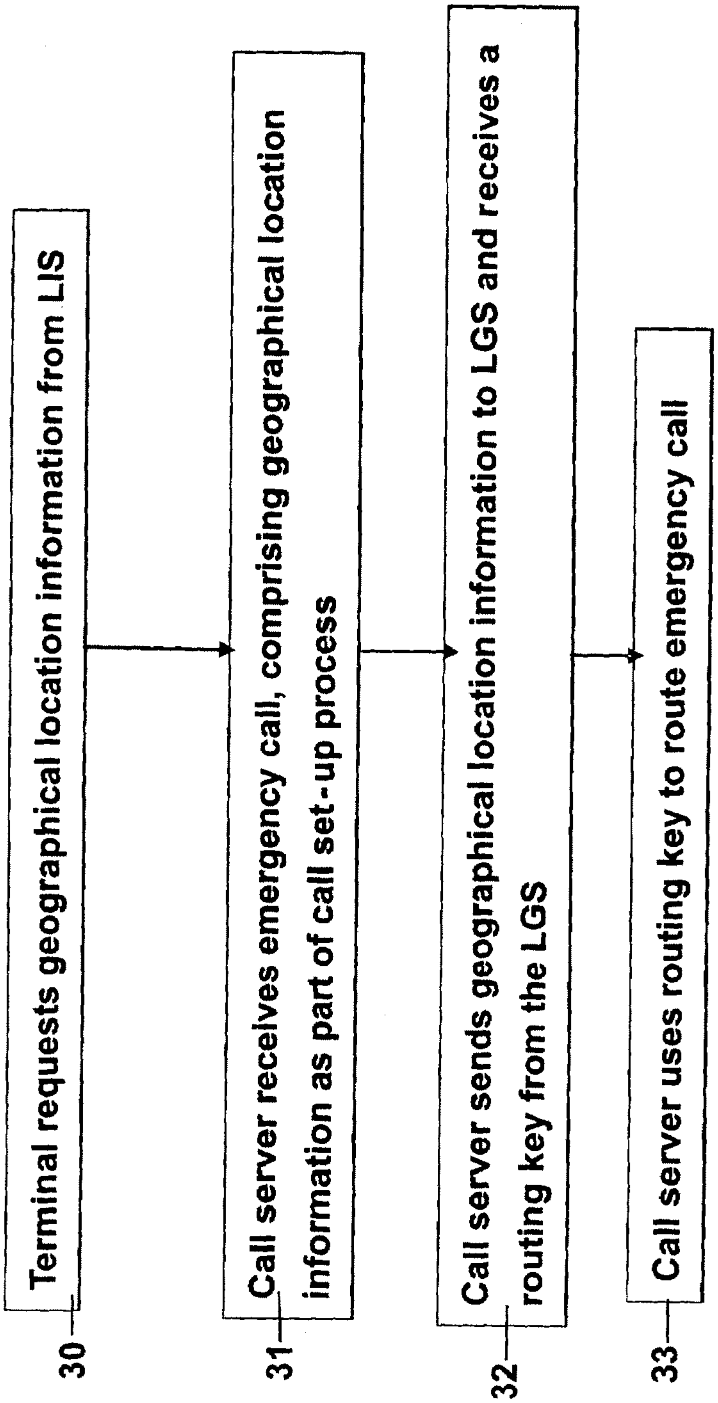

Terminal requests geographical location information from LIS
30
Call server receives emergency call, comprising geographical location information as part of call set-up process
31
Call server sends geographical location information to LGS and receives a routing key from the LGS
32
Call server uses routing key to route emergency call
33

Fig. 2

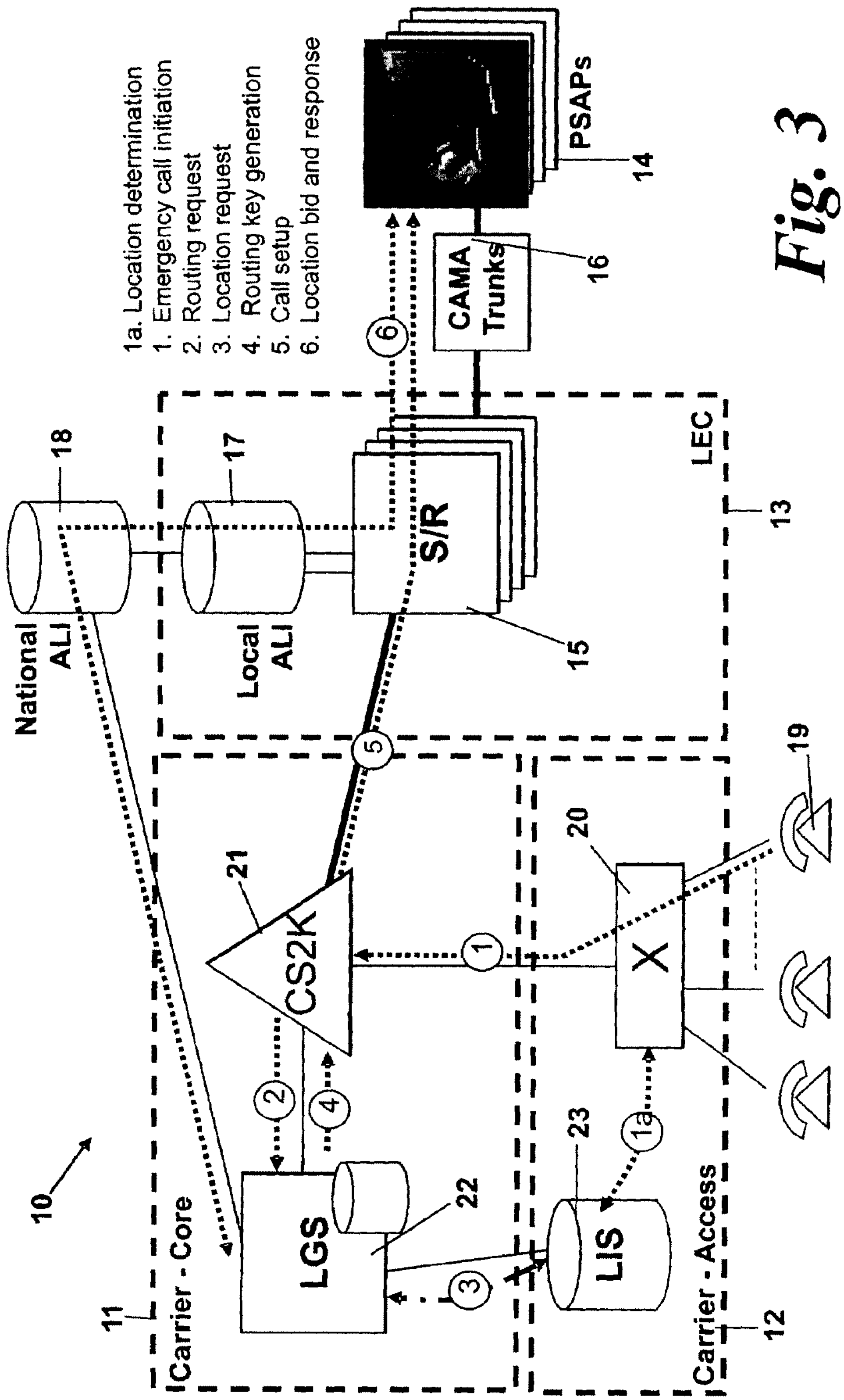

1a. Location determination
1. Emergency call initiation
2. Routing request
3. Location request
4. Routing key generation
5. Call setup
6. Location bid and response
PSAPs
14
CAMA Trunks
16
National ALI
18
Local ALI
17
S/R
15
LEC
13
5
6
CS2K
21
1
X
20
19
LGS
22
2
4
3
LIS
23
1a
Carrier - Core
11
Carrier - Access
12
10

Fig. 3

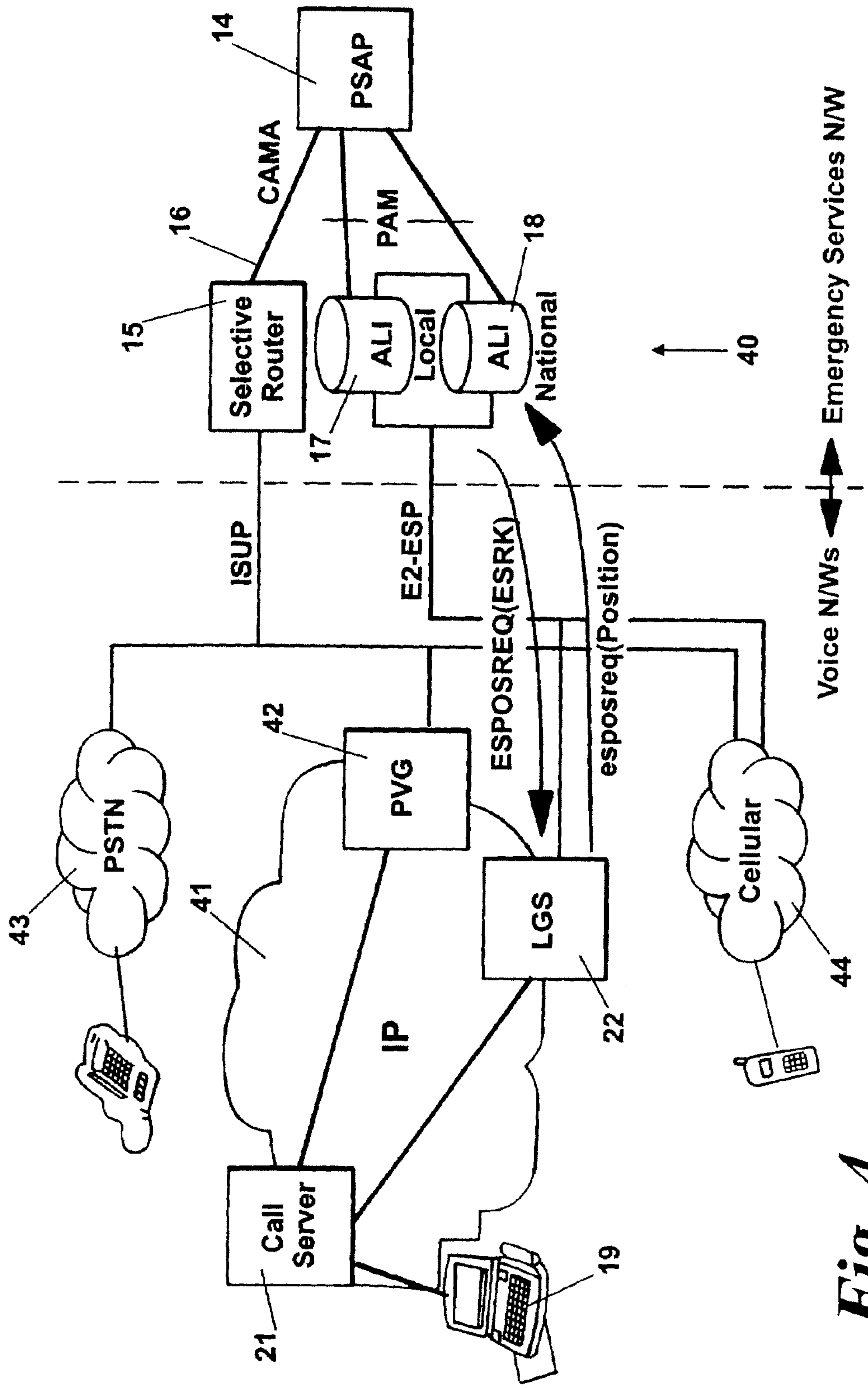

PSAP
14
CAMA
16
PAM
18
Selective Router
15
ALI
Local
ALI
National
17
40
Emergency Services N/W
ISUP
E2-ESP
ESPOSREQ(ESRK)
esposreq(Position)
Voice N/Ws
42
PVG
PSTN
43
41
LGS
Cellular
44
IP
22
Call Server
21
19

Fig. 4

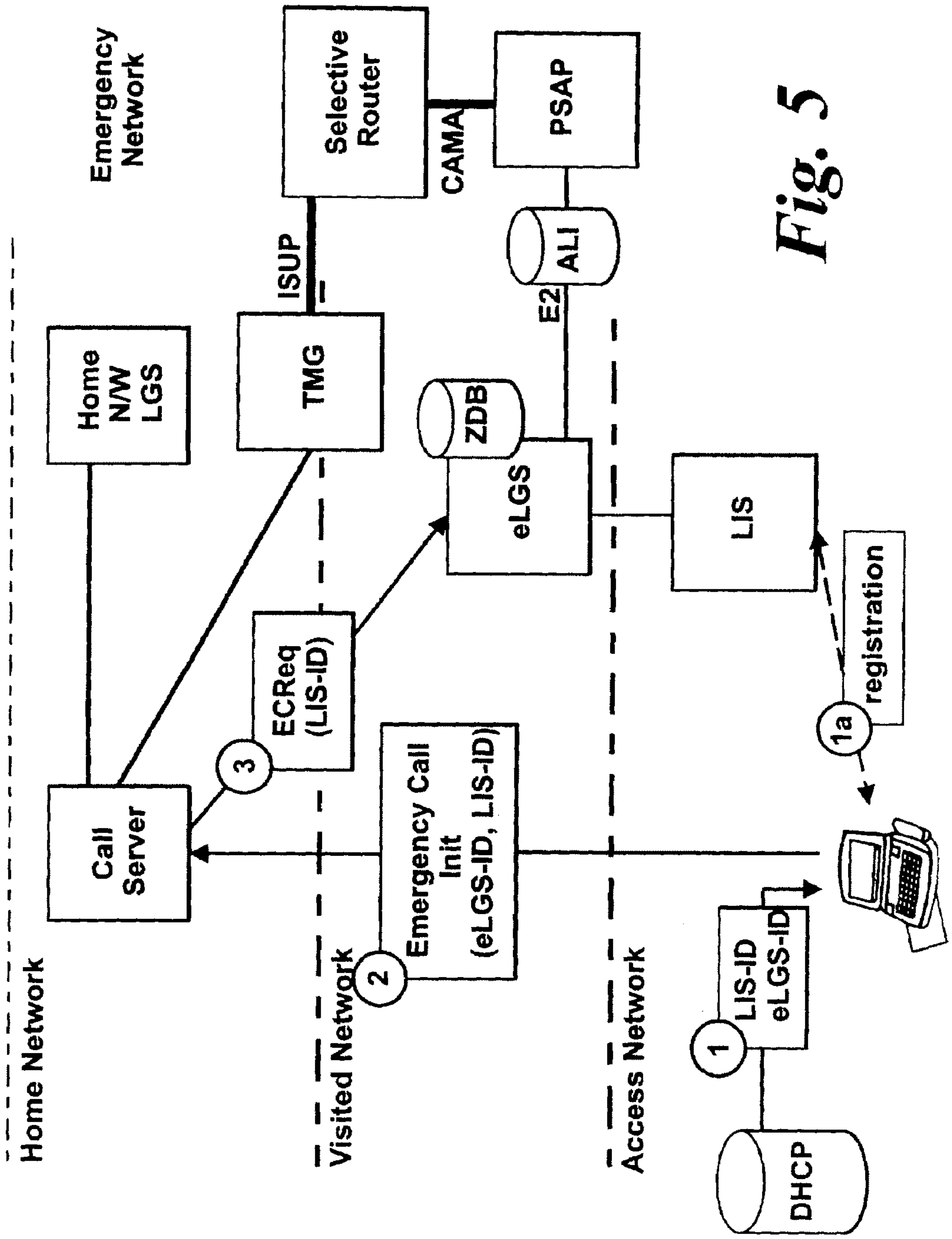

Emergency Network
Selective Router
PSAP
CAMA
ISUP
TMG
Home N/W LGS
ZDB
ALI
E2
eLGS
LIS
registration
1a
ECReq (LIS-ID)
3
Call Server
Emergency Call Init (eLGS-ID, LIS-ID)
2
LIS-ID eLGS-ID
1
DHCP
Home Network
Visited Network
Access Network

Fig. 5

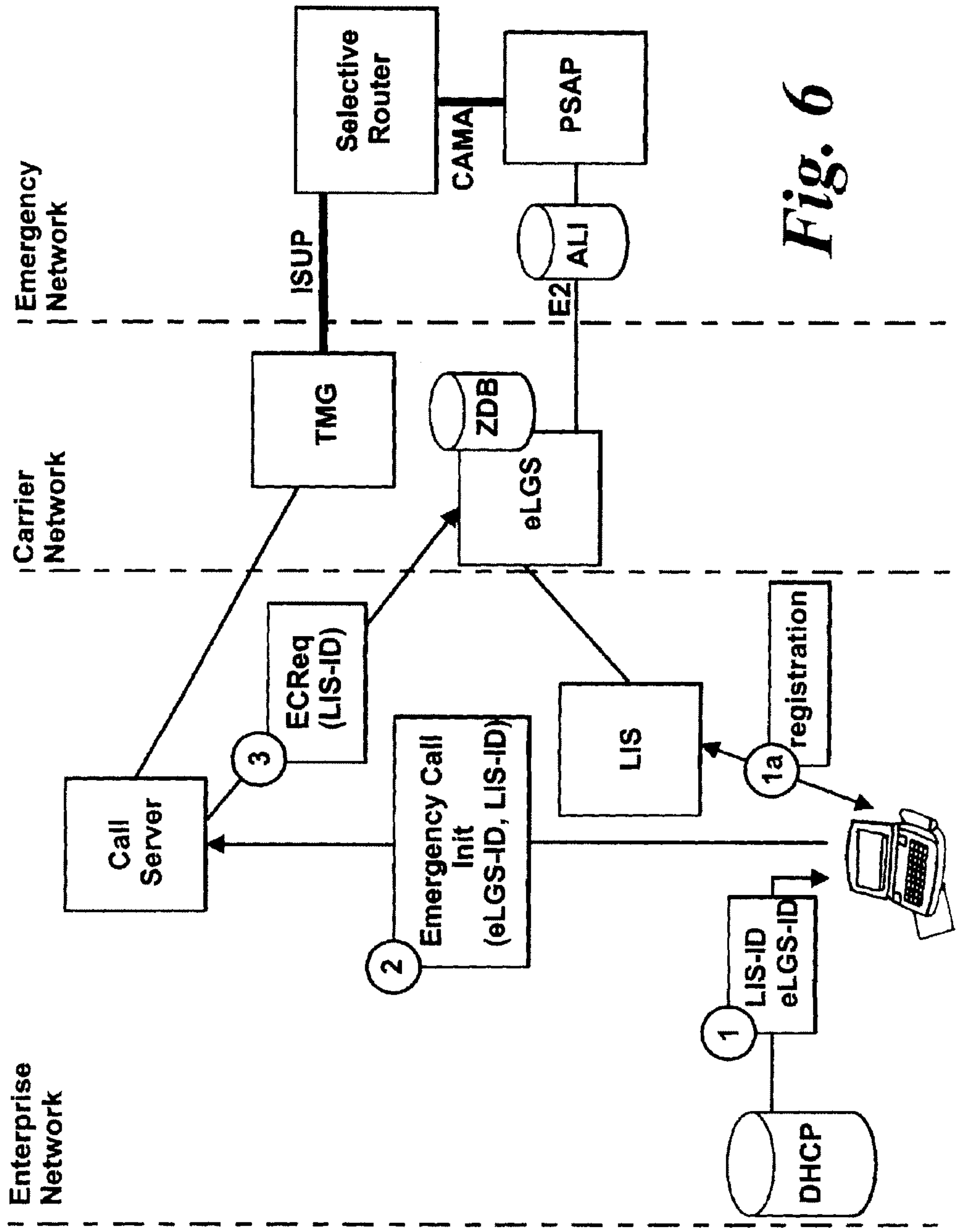
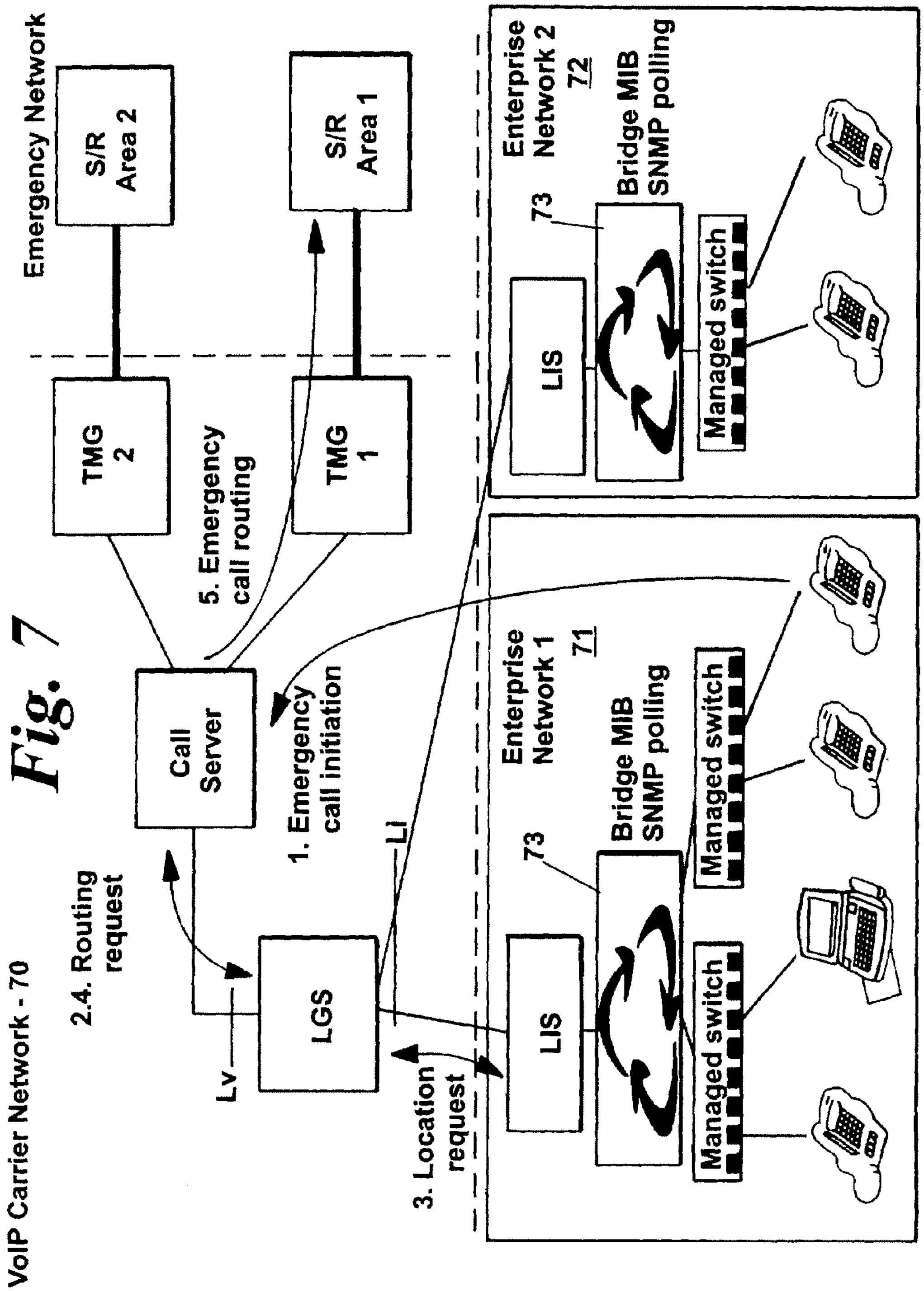

Emergency Network
Carrier Network
Enterprise Network
Selective Router
PSAP
CAMA
ISUP
ALI
E2
TMG
ZDB
eLGS
ECReq (LIS-ID)
3
Call Server
Emergency Call Init (eLGS-ID, LIS-ID)
2
LIS
1a
registration
LIS-ID eLGS-ID
1
DHCP

Fig. 6



VoIP Carrier Network - 70
Emergency Network
S/R Area 2
S/R Area 1
TMG 2
TMG 1
5. Emergency call routing
Call Server
1. Emergency call initiation
2.4. Routing request
LGS
Lv
Ll
3. Location request
Enterprise Network 1
71
Enterprise Network 2
72
73
Bridge MIB SNMP polling
LIS
Managed switch

Fig. 7

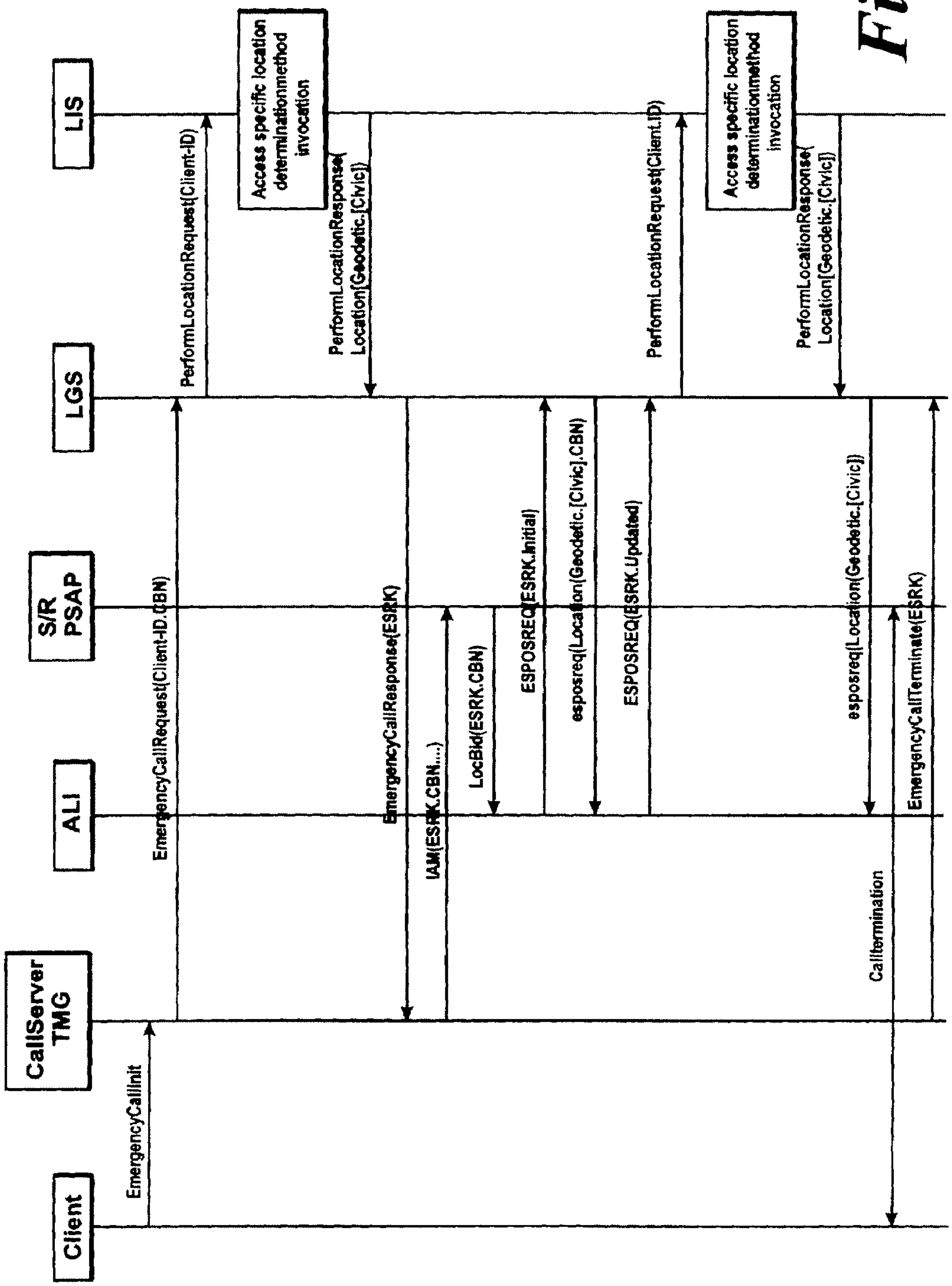

Client
CallServer TMG
ALI
S/R PSAP
LGS
LIS
EmergencyCallInit
EmergencyCallRequest(Client-ID.CBN)
PerformLocationRequest(Client-ID)
Access specific location determinationmethod invocation
PerformLocationResponse(Location[Geodetic.[Civic])
EmergencyCallResponse(ESRK)
IAM(ESRK.CBN...)
LocBid(ESRK.CBN)
ESPOSREQ(ESRK.Initial)
esposreq(Location(Geodetic.[Civic].CBN)
ESPOSREQ(ESRK.Updated)
PerformLocationRequest(Client.ID)
Access specific location determinationmethod invocation
PerformLocationResponse(Location[Geodetic.[Civic])
esposreq(Location(Geodetic.[Civic])
Calltermination
EmergencyCallTerminate(ESRK)

Fig. 8

DETERMINING THE GEOGRAPHICAL LOCATION FROM WHICH AN EMERGENCY CALL ORIGINATES IN A PACKET-BASED COMMUNICATIONS NETWORK

RELATED APPLICATION

This application is the non-provisional filing of provisional application No. 60/548,746, filed Feb. 27, 2004.

The present invention relates to a method and apparatus for determining the geographical location from which an emergency call originates in a packet-based communications network. The invention also relates to a method and apparatus for providing a routing key for routing an emergency call from a packet-based communications network node to an emergency services network node.

BACKGROUND TO THE INVENTION

There are a number of particular problems in dealing with emergency calls that do not arise for regular calls. For example, in order that emergency service vehicles or other assistance can be dispatched to the correct destination promptly, accurate information about the location of the caller is needed. Previously, in conventional switched telephone networks, it has been possible to provide the caller location information relatively easily because telephone handsets are typically fixed in particular locations. Static database entries can then be made in a database accessible to the emergency services associating for example, a subscribers' home address and telephone number. However, for mobile communication systems and also for nomadic systems use of such static database entries is not possible because the location of a communications terminal varies over time.

Another problem concerns routing emergency calls to the correct destination. For regular calls this is not such an issue because the caller enters specific details of the required call destination. However, for emergency calls a universal code is used such as 911 in North America and 112 in Europe. This universal code cannot be used to identify the destination of the call. Generally, an emergency call needs to be routed to a particular geographical answering point for servicing. This answering point is often referred to as a Public Safety Answering Point (PSAP). The jurisdiction for emergency services answering points is typically quite small, for example, at the county level in the USA. This information about the location of the caller is needed to determine which emergency services answering point to route the call to. Misrouting of calls to the wrong answering point leads to costs in transferring calls, impacts reliability, and leads to delays which are significant in life threatening situations. Previously, in conventional switched telephone networks, this location information was relatively easy to obtain because static database entries could be used as mentioned above. However, this is not possible for mobile and nomadic communications systems.

One proposal has been to update or refresh the database entries every 24 hours. However, this approach cannot cope with situations where a user terminal changes location more than once a day. Also, changes to the existing emergency services network infrastructure are required in order to enable the database to be updated daily.

More detail about how existing voice networks interface to the emergency services network is now given. The primary existing voice networks that do interface to emergency services are the PSTN (public switched telephone network) as served by LECs (local exchange carriers) and the various mobile networks operated by the cellular carriers.

The emergency services network, from this perspective, can be regarded as being made up of Selective Routers (SRs), Automatic Location Identification (ALI) databases, both local and national, and the Public Safety Answering Points (PSAPs) themselves with their various CAMA (centralized automatic message accounting), and other, trunk connections and various data connections for querying the ALIs. Of course, beyond these network elements are the public safety organisations themselves (Police, Fire, Ambulance) and the communications networks that support them.

The location of the subscriber, who is dialing emergency services, is used for two key purposes. The first is routing of the call, ultimately to the right PSAP, and the second is in the delivery of the location, for display, to the PSAP operator in order that emergency response units can be dispatched to the correct location.

In wireline voice networks, there is an association between the phone number of the subscriber (The Calling Line Identifier—CLID) and that subscriber's location. This is generally, the home address of the subscriber as maintained by their local exchange carrier. In this case, the CLID becomes a ready-reference to location.

Similarly, the incoming line to the local exchange switch and the switch itself provides an explicit indication of the appropriate routing of 911 calls. This permits the local exchange to work from a static configuration in terms of selecting the outgoing trunk on which to place the call so it goes to the correct selective router. The selective router, in turn, can use the same static association and CLID information to ensure that the call is routed to the correct serving PSAP for the subscriber's address.

In cellular systems, the association between the subscriber's location and their CLID is lost. Being, by definition, mobile a cellular subscriber can be anywhere within the wireless network's area of coverage. Similarly, there is no physical wired line corresponding to the source of the call from which to associate a route to the correct destination. In cellular networks, however, there is a physical serving cell from which the call is initiated. The geographic granularity of these cell locations is generally sufficiently fine for the mobile switch to determine the correct trunk route to a corresponding selective router. In many cases, this also provides sufficient accuracy for the selective router to determine which PSAP the caller should be connected with.

It is an internal procedure for the mobile switch to associate an outgoing trunk route with a serving cell. However, some signaling is required for an MSC (mobile switching center) to pass this same information along to the selective router so that it can determine the correct PSAP. The TR45 standard, J-STD-036 "Enhanced Wireless 9-1-1 Phase 2", Telecommunications Industry Association, 2000, defines mechanisms for doing this. The routing information is passed to the selective router in the ISUP (ISDN user part) call setup signaling in one or other newly defined parameters called the Emergency Services Routing Digits (ESRD) or the Emergency Services Routing Key (ESRK). The selective router examines the value of the ESRD/ESRK parameter in the call setup signaling and routes the call to the correct PSAP based on this value.

Note that there are circumstances where cell boundaries can span the boundaries of PSAP catchment areas. In this case, and ESRD or ESRK determined from a serving cell may not provide a reliable indication of a route to the correct

PSAP. Both ANSI-41 (generally TDMA, and CDMA) and 3GPP (generally GSM, EDGE, and UMTS) cellular networks have identified functionality to address this. In ANSI-41 networks a functional element known as a Coordinate Routing Database (CRDB) is defined. The network can consult the CRDB and, based on the geographic location of the caller (determined by different positioning technologies such as forward link trilateration, pilot strength measurements, time of arrival measurements, etc.), it will return an appropriate value of the routing parameter. As long as the geographic location is an improvement in accuracy over the cell location, this mitigates the problem of misrouted calls. Similarly 3GPP networks allow the switch to request a refined routing key value from the Gateway Mobile Location Center (GMLC) based on the geographic location of the caller.

The second, independent, area in which location comes into play in E911 calling is the display of the caller's location to the PSAP operator. The need for this is that the PSAP operator can facilitate more rapid despatch of the emergency service response units if the network can deliver the location rather than relying on getting this information from the caller—particularly where the caller may be unable to provide this information.

In a wireline voice network, necessary subscriber (or, at least, calling line) address information is stored in a database known as an Automatic Location Identification, or ALI, database. On receipt of an emergency call and, armed with the caller's CLID, the PSAP is able to query this database and receive, in return, the street address (also known as a civic address) information associated with the CLID. The physical interface over which the PSAP makes this query is variable. It may be an IP based interface over dial-up or broadband or it may be made over an X.25 packet interface. Similarly, the ALI may physically be co-located within the LEC and selective router, or it may be a remote national ALI handling the request directly or in tandem from the local ALI. Similarly, the protocol may vary but one known as PAM (PSAP to ALI message specification) is in common usage. These details are contained within the emergency network itself and not generally a concern of the larger voice network on the far side of the selective router.

In a cellular network, the same level of detachment with respect to this function is not possible. To begin with, the location of the caller is variable both initially and during the period of an emergency call. It is no longer possible to rely on a static database of location information that can provide an address against a CLID. It now becomes necessary for the PSAP to be able to request a dynamic location both for the initial position of the caller but also for any changes during the call. In addition, a civic address may no longer be pertinent to the location of the caller. By nature, cellular networks cover wide and varying types of territory. A conventional street address may no longer apply to a caller's location. Indeed, they may not even be in or by a street as the term is commonly understood. For this reason, a more universal reference system for location needs to be used. The solution generally adopted and, once more defined in J-STD-036 as referenced above, is to use geospatial co-ordinates—or latitude and longitude—as defined in the WGS-84 coordinate system (Military Standard WGS84 Metric MIL-STD-2401 (11 Jan. 1994): "Military Standard Department of Defence World Geodetic System (WGS)").

While J-STD-036 does define mechanism whereby this geospatial location can be delivered in the ISUP call setup signaling, it can be generally acknowledged that PSAPs do not support the necessary signaling interfaces nor customer

premises equipment to receive and display this information. Also, there is no mechanism whereby an updated location can be delivered in the ISUP signaling. For these reasons, J-STD-036 identifies a new interface that the emergency network can use to query the cellular network. This interface is assigned the identifier of E2 and both J-STD-036 and NENA "NENA Standard for the Implementation of the Wireless Emergency Service Protocol E2 Interface" define a protocol which can be used over this interface called the emergency services protocol.

On receipt of an emergency call arising from a cellular network, the PSAP can initiate, via the serving ALI, a request on the cellular network to provide the geodetic location of the caller. This request is made over the E2 interface in a message called the EPOSREQ (Emergency Position Request) with the response message identified as the esposreq. The location of the caller is determined by positioning capabilities native to the cellular network itself and different systems of network measurement, triangulation, or special handset capabilities such as GPS (Global Positioning System) are used.

As described above, the network mechanisms and procedures defined in JSTD-036 are around the provision of a geodetic (latitude and longitude) type location for the caller. This obviously implies a capability on the part of the PSAP to display location information of this type to the PSAP operator. There is also consideration supported in the E2 interface messaging that allows the delivery of civic address type information.

One application of this facility is in the support of PSAPs which are not equipped with the capability to receive and display geodetic type location information. This is part of what is often referred to as a Phase 1 E911 capability for cellular networks. Enhanced 911 calling was introduced in two phases into the cellular and emergency services networks. Phase 2 defined the capabilities for delivering, generally more accurate, geodetic location information from the network. Phase 1 was generally targeted at providing location information to the accuracy of a serving base station location but, perhaps more importantly, that location information is delivered to the PSAP as a more conventional street, or civic, address associated with that base station. Depending on the nature of the PSAP, the ALI may provide the geodetic position and/or the phase1 civic address type information in response to the location bid.

Just as cellular networks have specific characteristics that result in new considerations for E911 compared to conventional wireline voice networks, so too do IP based voice (VoIP) networks. VoIP network users have much in common with cellular network users in that there is no specific physical point of connection which dictates their identity. Just as a cellular phone can attach to the network anywhere that there is a point of coverage, so too can an IP based phone client attach to an IP network at many and varied points and take advantage of the voice service. From this perspective, it becomes necessary to view VoIP clients as essentially nomadic or even fully mobile to ensure that all usage scenarios are covered. For certain, many VoIP clients may be relatively static in terms of location (for example, a conventional form factor desktop phone with integrated VoIP client software will tend to be as stationary as any conventional wireline desktop phone) however, this situation is not explicitly predictable by the network, so an architecture that addresses mobility ensures that all usage scenarios are covered.

In terms of emergency call routing, the VoIP network introduces some additional challenges over wireline or cel-

lular networks. In particular, the access network associated with a VoIP network can be highly distended. That is to say, in wireline the phone is tied to the specific local switch by the incoming line, in cellular the mobile switch has specific knowledge of the serving cell which has some degree of geographic association with that switch. But, in VoIP, the client may be attached to the network in another city, state, or, even, country than the one in which the serving call server is located. There is not an immediate association to location that the call server can use to directly determine a route to a selective router before, even, the correct PSAP can be selected.

Similarly, in terms of location delivery and display, a VoIP client may be appropriately identified by a street address, being on a relatively static access point, or it may be more appropriately identified against a geodetic location, as in the case of a VoIP client connected by a wide area broadband wireless network.

OBJECT TO THE INVENTION

The invention seeks to provide a method and apparatus for determining the geographical location from which an emergency call originates in a packet-based communications network which overcomes or at least mitigates one or more of the problems mentioned above.

The invention also seeks to provide a method and apparatus for providing a routing key for routing an emergency call from a packet-based communications network node to an emergency services network node which overcomes or at least mitigates one or more of the problems noted above.

Further benefits and advantages of the invention will become apparent from a consideration of the following detailed description given with reference to the accompanying drawings, which specify and show preferred embodiments of the invention.

SUMMARY OF THE INVENTION

According to an aspect of the present invention there is provided a method of providing a routing key for routing an emergency call from a packet-based communications network node to an emergency services network node in a switched telephone network, said method comprising the steps of:

- receiving information about the geographical location from which the emergency call originates;
- generating a routing key on the basis of the received information and pre-specified information about geographical locations served by particular emergency service network nodes.

This provides the advantage that an emergency call can be routed using the routing key to an appropriate emergency services network node. This is achieved in a packet-based network without the need to access information from the emergency services network. Thus an existing emergency services network can be used without the need for modification.

Preferably the method comprises storing said generated routing key together with the received information about geographical location. The method also comprises providing the stored information to an automatic location identification (ALI) database. In this way the geographical location information is made available to an existing emergency services communications network comprising an ALI. The emergency services network is then able to display that information and use it to dispatch emergency services vehicles.

According to another aspect of the present invention there is provided a packet-based communications network node for providing a routing key for routing an emergency call from the packet-based communications network to an emergency services network node in a switched telephone network, said node comprising:

- an input arranged to receive information about the geographical location from which the emergency call originates;
- a processor arranged to generate a routing key on the basis of the received information and pre-specified information about geographical locations served by particular emergency service network nodes.

According to another aspect of the present invention there is provided a method of routing an incoming emergency call in a packet-based communications network to an appropriate emergency services answering point in a switched telephone network, said method comprising:

- at a call server, receiving the emergency call;
- at a location gateway server, receiving a geographical location from which the call originated and using that to generate a routing key;
- at the call server, routing the emergency call using the generated routing key.

Preferably a location information server is used to provide the geographical location information. This provides the advantage that the location gateway server need not be concerned with the particular methods used to determine the geographical location information.

Also, the routing key is determined and delivered dynamically within the life of the emergency call. This is achieved by using the location information server to provide the geographical location information as and when needed. This reduces and need for static information to be retained in the network including an emergency services network. In addition, it is possible to deal with nomadic entities and mobile entities whose geographical location changes over time.

In a preferred embodiment the location gateway server interfaces to the emergency services network using a known interface protocol. This enables the present invention to be used with existing emergency services equipment that already operates the specified interface protocol. This reduces costs and the need for modification of network equipment.

The invention also encompasses computer software for implementing any of the methods described above and herein.

The preferred features may be combined as appropriate, as would be apparent to a skilled person, and may be combined with any of the aspects of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

In order to show how the invention may be carried into effect, embodiments of the invention are now described below by way of example only and with reference to the accompanying figures in which:

FIG. **1** is a schematic diagram of a packet based communications network comprising a location gateway server;

FIG. **2** is a flow diagram of a method of operating a call server to route an emergency call;

FIG. **3** shows the communications network of FIG. **1** with a connection between the location gateway server and location information sever;

FIG. **4** is a schematic diagram of another embodiment of the communications network of FIG. **1**;

FIG. **5** is a schematic diagram of a communications network comprising a DHCP (Dynamic Host Configuration Protocol) server suitable for use in an embodiment of the invention;

FIG. **6** is a schematic diagram of an enterprise client and call server using a carrier Location Gateway Server (LGS) and Trunk Media Gateway (TMG) to route emergency calls into an emergency network;

FIG. **7** is a schematic diagram of a carrier voice over internet protocol (VoIP) deployment serving enterprise customers with network based VoIP service from legacy clients connected via conventional managed switches;

FIG. **8** is a message sequence chart for several embodiments of the invention.

DETAILED DESCRIPTION OF INVENTION

Embodiments of the present invention are described below by way of example only. These examples represent the best ways of putting the invention into practice that are currently known to the Applicant although they are not the only ways in which this could be achieved.

The term "geographical location information" is used to refer to information about the physical position of an entity in the physical environment as opposed to a communications network address. For example, it comprises a civic address, postal address, street address, latitude and longitude information or geodetic location information.

The term "nomadic communications system" is used to refer to a communications network in which user terminals can access the network from different, geographically separated, network access points without the need for modification of the terminal in order to access the network from those different access points.

FIG. **1** is a schematic diagram of a packet-based communications network **10** with a core network region **11**, an access network region **12**, and a local exchange region **13**.

A plurality of public safety answering points (PSAPs) **14** are shown, each being for serving a different geographical region as known in the art. Each PSAP has an associated selective router **15** which is a switch for routing calls, location information and other details to the PSAP. Each selective router **15** is connected to its associated PSAP by a trunk **16** or other suitable communications link as known in the art.

Each selective router **15** is linked via the communications network **10** to a local automatic location identification (ALI) database **17**. This database comprises pre-specified information about a geographical address associated with each customer or user account and details of an identifier for a communications terminal for that customer account. Only information about customer accounts with geographic addresses local to the particular selective router **15** are stored in the local ALI **17**.

A national ALI **18** is also provided. This comprises pre-specified information about which geographical regions each local ALI **17** serves. For example, details of every valid postal address in the USA are stored and each address is associated with a particular local ALI **17** and selective router **15**.

The local ALIs **17**, selective routers **15**, national ALI **18**, PSAPs **14** and trunks **16** are all known in the art of conventional switched telephone networks. An advantage of the present invention is that this emergency service network infrastructure is reused without the need for modification. The existing emergency service network infrastructure is

integrated or connected to the packet-based communications network core **11** as using media gateways of any suitable type as known in the art.

Communications terminals **19**, also referred to as clients, which are of any suitable type, are connected to a switch **20** in the access part **12** of the communications network **10**. The terminals **19** are either physically connected to the network or connected via a wireless link. The switch is connected via the network to a call server **21** in the core of the network **11**. Only one call server **21** and switch **20** are shown for reasons of clarity, however, many switches **20** are typically served by one call server **21** and there can be a plurality of call servers **21**.

When a communications terminal connects to the network **10** on start-up of the terminal, or if the terminal is newly connected, then a registration request is sent to the call server **21** for that region of the network **10**. This process is known in the art. The registration process involves the terminal sending, via the switch **20**, details of its network address. The call server **21** is then able to keep track of all the terminals **19** under its remit.

In the present invention a location information server (LIS) **23** is provided. FIG. **1** shows the LIS in the access part of the network **12** although it can also reside in an Enterprise network or an access network for residential services. The LIS can also be split into two entities: probes and a main server, with the probes in an Enterprise network for example and the main server in an access network.

The LIS is arranged to detect terminals connecting to the network and determine their geographic locations. The LIS passes this information to the terminals when requested and the terminals pass it on to the call server. In addition, the LIS is able to pass the geographic location information to another entity, a Location Gateway Server (LGS) as described in more detail below with reference to FIG. **3**. In that case the LGS polls the LIS for the location information.

As mentioned above the LIS determines geographic location of terminals. It does this in any suitable known manner. For example, it comprises or has access to a wiremap. This wiremap comprises details of network addresses of access ports in the access network served by the call server **21** and geographic location details associated with each network address. For example, a building address and a particular quadrant of that building. This information is pre-configured at the LIS, for example, by service providers or other network administrators. Thus when a terminal **19** is connected to a particular access port in the access portion of the network **12** there is a network address associated with that port and at the LIS geographic location details associated with the same port or network address. However, it is not essential for the LIS to use a wiremap. Any suitable type of positioning technology can be used. An advantage of using an LIS in this way is that the LGS need not be concerned with the nature of the positioning technology used.

The core network **11** also comprises a location gateway server (LGS) **22** connected to the call server **21** and also linked to the national ALI **18**. The LGS is a novel network entity for use in the present invention. The LGS **22** is arranged to determine routing keys also known as emergency services routing keys (ESRKs). A routing key is used by the call server **21** to route an incoming emergency call to an appropriate selective router **15** and PSAP **14**. In order to determine the routing keys the LGS operates in conjunction with the LIS **12** and national ALI **18**.

In a first embodiment of the present invention the LIS is arranged to detect when a terminal newly starts up or connects to the network. The LIS determines a geographic

location for that newly connected terminal using any suitable method as known in the art. For example, the LIS accesses a wiremap as mentioned above and uses that together with a network address of the terminal to determine an associated geographic location. Alternatively, a global positioning system is used or an emergency caller specifies his or her own location.

In the event that an emergency call is made from one of the terminals **19** that terminal **19** sends a request via switch **20** to the LIS **23** for its geographical location (see box **30** of FIG. **2**). That geographic location information is returned and sent by the terminal **19** to the call server **21** as part of the call set-up process (see box **31** of FIG. **2**).

The call server then sends the geographical location information to the LGS. For example, this information is sent in the form of a subscriber location report (SLR) and comprises a call back number for the emergency caller as well as the geographical location information (see box **32** of FIG. **2**.) However, this is not essential, any suitable form of message can be used to send the geographical location information.

The LGS uses the geographical location information to determine the relevant selective router **15** and PSAP **14** and generates an appropriate routing key. The LGS stores the geographical location information together with the routing key in a cache or other suitable memory. The routing key is made available to the call server (see box **32** of FIG. **2**) which then routes the emergency call to the specified selective router **15** (see box **33** of FIG. **2**) via a media gateway. The selective router **15** then delivers the emergency call to the appropriate PSAP together with the routing key generated by the LGS. In some cases instead of a routing key a pseudo-ANI generated by the selective router and the local ALI is sent to the PSAP instead of the routing key.

In the method described above with reference to FIGS. **1** and **2** the LIS is arranged to provide geographical location information to terminals which then provide it to the call server. However, in some situations the location information does not reach the LGS. For example, if there is an error in transmission and packets are dropped. Also, there are situations in which the user terminal is a wireless device that is moving. In that case it may not be possible for the LGS to keep up to date with the rapidly changing location of the mobile terminal.

A second method for enabling the location information to reach the call server is therefore proposed and is now described with reference to FIG. **3**. This method is preferably used in conjunction with that of FIG. **1** although that is not essential; the two methods can be used independently of one another. Using the methods independently, although not as fool-proof as using them together, is acceptable in some cases. For example, where location information is available via another means. This other means can be for example, explanation from the emergency caller him or herself or a separate global positioning system device of the emergency caller.

FIG. **3** shows the same components as in FIG. **1** and the same reference numerals are used as appropriate.

In this second method the LGS **22** polls or queries the LIS **23**. For each port at the switch **20** to which a terminal is connected the LIS determines an associated geographical location as described above with reference to FIG. **1**. Thus in this second method the LGS polls the LIS rather than waiting for geographic information sent from the call server. The LGS is also able to do both these things; that is, poll the LIS for the geographic information and receive it from the call server.

Consider the situation when an emergency call is made from one of the terminals **19**. This call reaches the call server **21** as known in the art and the call server **21** receives an identifier of the calling terminal as part of the call process. The call server then sends a message to the LGS requesting geographical location information for the emergency calling terminal. The message comprises a subscriber location report or any other suitable type of message. The message comprises the identifier of the calling terminal as well as details of the LIS associated with the call server **21**.

The LGS itself does not have the geographical location information requested and so it queries the LIS for that information using the identifier of the calling terminal.

As in the method described with reference to FIG. **1**, the LGS uses the geographical location information to determine the relevant selective router **15** and PSAP **14** and generates an appropriate routing key. The method then proceeds as described above with reference to FIG. **1** such that the emergency call is routed to the appropriate PSAP and the location information is also delivered to the PSAP.

Thus both the first and second methods described above involve using the LIS to determine the location from which an emergency call originates. In the first method the LIS sends this geographical information to a terminal which sends it to the call server during call set up. In the second method the LGS actively polls the LIS for the geographical location information.

The methods described thus far enable an emergency call to be routed to the appropriate PSAP. In order for the PSAP to also obtain the geographical location information of the emergency caller an interface is provided between the LGS and the emergency services network. This is now described with reference to FIG. **4**.

FIG. **4** is a schematic diagram of another embodiment of the network of FIG. **1**. The same reference numerals are used as appropriate. An emergency services network **40** as known in the art of conventional switched telephone networks is connected to a packet-based communications network **41** via one or more media gateways **42**. A conventional public switched telephone network **43** is also connected to the emergency services network **40** via any suitable type of interface such as an ISDN User Part (ISUP) interface, as is a conventional cellular network **44**. In a preferred embodiment of the present invention an interface between the LGS **22** and the emergency services network **40** is provided using the same method as used to interface between location gateway entities in a cellular network and an emergency services network. At least part of the present invention lies in the realisation that an interface from a cellular network can be reused to integrate a packet-based network and a conventional emergency services network. Preferably the E2 interface standard defined in TR45 J-STD-036 "Enhanced Wireless 9-1-1 Phase 2" Telecommunications Industry Association, 2000 is chosen although any other suitable interface method can be used. The aforementioned document is incorporated herein by reference.

FIG. **4** shows the LGS **22** connected to the local and/or national ALI of the emergency services network **40** using an E2 emergency services protocol (ESP) as mentioned above. This ESP allows the emergency network to make a request for a caller location which is then delivered for display to a PSAP operator.

For example, the PSAP sends a query to the local ALI **17**. In FIG. **4** this query is referred to as ESPOSREQ (ESRK). The query contains details of the routing key and requests the associated geographical location information.

The local ALI **17** forwards the query (also known as a bid) to the national ALI **18** which in turn forwards the query to the appropriate LGS **22**. The LGS has previously stored the routing key together with the geographical location information and so it is able to return the geographical location information to the national ALI. This is shown in FIG. **4** as esposreq (Position). From there it is returned, via the local ALI, to the PSAP.

At the LGS, the cached routing key and location information are cleared from memory when appropriate. For example, after a pre-specified time interval or at the end of the emergency call. In the latter case, the call server **21** is arranged to send a message indicating call terminal to the LGS. The message is of any suitable form such as a subscriber location report.

More detail about particular examples of the present invention is now given.

Emergency Call Routing

For the sake of simplicity, the following discussion is based on the assumption that the Selective Router will use an ESRK provided in the ISUP call setup (IAM—Initial Address Message) to select the correct outgoing CAMA trunk for the corresponding serving PSAP. However, it is not essential to use an ESRK. Any suitable key such as an ESRD can be used instead.

Trunk Media Gateway (TMG) to Selective Router Routing

Opening up the VoIP network cloud, we can see that an emergency call needs to be delivered into the wireline voice network in order to enter the existing emergency services network. In VoIP networks, this is done by transiting the call out of the IP network and into wireline network via a Trunk Media Gateway (TMG).

Since there are no dialed digits that can be used to effect routing of the emergency call (911 does not identify a unique destination), it is necessary for the TMG selected to have direct ISUP trunking capability to the selective router(s) that it supports routing to. To reuse the cellular mechanism for call routing, the TMG needs to provide a unique ESRK to the selective router. That is, the TMG ISUP signaling preferably supports the inclusion of this parameter in the IAM message. Further, if the TMG has outgoing trunks to more than one selective router, it needs to be instructed as to which trunk to select based on the ESRK. That is, in the absence of routing based on dialed digits, the TMG needs to be told which outgoing voice trunk and ISUP signaling destination to select based on the value of the ESRK for that call. This implies a routing table that the network will use to ensure that the TMG is appropriately directed.

Call Server to TMG Routing

Looking further back into the VoIP network cloud, we see that the VoIP call itself is under the control of a call server. This network entity provides at least the equivalent functionality of a wireline switch or a cellular mobile switching center. The call server is responsible for setting up the initial state associated with an emergency call and routing it to the correct destination.

As has been noted at each step, the dialed digits do not provide a definitive route to the destination and, as noted in the previous section, the TMG outgoing trunk needs to be selected based on the ESRK so the appropriate selective router is trunked to. Since the call is delivered to the TMG by the call server, it is the responsibility of the call server to provide this ESRK in the IP based call setup and corresponding trunk selection through the TMG.

Since the call server has the responsibility to select a TMG based on the ESRK, the existence of a routing table within the call server is implied. This table allows the call server to associate a TMG with a given ESRK value.

Location Based Emergency Call Routing

This section describes an example of how the call server determines the ESRK associated with final destination PSAP. This is addressed by the introduction of a new network entity called the Location Gateway Server (LGS). This network entity supports two key functions:

- On request from a call server, and given the identity of an emergency caller/client, it obtains the location of that client from the IP access network. For routing purposes, this location may be provided as a geodetic (latitude/longitude) location.
- Based on the location determined, and using a native spatial database capability which can identify an emergency services zone corresponding to a destination PSAP, it generates a unique and applicable ESRK value that will indicate a route to the correct serving PSAP.

A single message and response is defined between the call server and the LGS which is used by the call server to request the ESRK. These are the EmergencyCallRequest (ECR) and the ECResponse messages. The key parameters of the request and response are the client ID in the former and the ESRK in the latter.

A second message, ECTerminate, is also required to indicate the termination of the emergency call. The LGS maintains transient state information associated with emergency calls in progress. It needs to allocate an ESRK out of a pool of available numbers and it needs to be able to return the ESRK to this pool at the conclusion of the call. Thus, it is important for the call server to provide a message to the LGS indicating that the call is terminated. The ESRK associated with the call and provided in the call termination indication message provides the necessary state association for the LGS.

It is also possible for the call server to provide the initial location of the client in the message to the LGS. This is also useful in a situation where there is no LIS and clients/users specify their own location (e.g. picked from a menu).

Emergency Caller Location Delivery

The question of how an LGS determines the location of a client device is described later. Before looking into that question, the other aspect of location—the delivery of it to the PSAP operator—is examined.

As has been noted, the location of a VoIP client can be a transitory piece of information. As such, it is not adequate—as a general solution—to rely on a static data entry accessible by the emergency network and keyed against the CLID. As with cellular networks, the information associated with a subscriber should be determined, and is only valid, within the time that the call is active. Outside the period of duration of the emergency call, the emergency network stores no information and has no knowledge related to the identity or location of the subscriber.

In order to support Phase 2 E911 requirements, J-STD-036 defined the E2 interface between the ALI entities in the emergency services network and the location gateway entities (GMLCs and MPCs) in the connecting cellular networks. The emergency services protocol (ESP) supported over this interface was defined by both J-STD-036 and in the NENA publication mentioned above.

An embodiment of this invention teaches that this same E2 interface and ESP protocol specification be reused on the

LGS to support the delivery of location information associated with VoIP emergency calls.

The ESRK becomes a reference to the call in progress as well as being the routing indicator used in call setup. ESP allows the emergency network to make a request for a caller location which can then be delivered for display to the PSAP operator. The LGS already has the location information for the client since it was used to deliver the call routing information. By caching this location in conjunction with the ESRK call-in-progress, state, the LGS is able to provide this location information in the esposreq sent in response to a request made over the E2 interface by the emergency network.

Mid-Call Location Updates

Since cellular subscribers can, by definition, be mobile, the ESP semantics also support the ability for the emergency network to request an updated location for the caller. Using the same call identifier (e.g. the ESRK) as was used to request the location initially, the same ESPOSREQ message is used to request an updated location. That is, there is a parameter in this message to indicate which type of location—initial or updated—that the emergency network would like. If an updated location is required, the cellular network knows that it should utilize its resources to see if a more up to date location is available.

This same mechanism is used in an embodiment of the present invention for the VoIP network. While in initial deployments, the IP access networks may only return relatively static locations (e.g. from switch port wire mappings), future deployments will be able to exploit advanced positioning technologies that can track a mobile IP device, just as they can a mobile cellular device today. Since the semantics for requesting an updated location are already supported on the E2 interface, there will be no changes necessary to the emergency network in order for it to exploit this tracking capability.

Civic Address and Geodetic Location Support

The introduction of Phase 2 E911 support for cellular emergency callers introduced the concept, and the precedent, that the location of the caller may actually be provided to the PSAP as a geodetic location. This has necessitated changes to PSAPs such that to be Phase 2 capable they need not only the ability to display a location in this format to an operator but also that these PSAPs have the necessary procedures and policies in place to relay location information in this form to emergency response teams and be able deal with accuracy that can vary below 100 meters at the 67th percentile and approach arbitrary levels of inaccuracy for the other ⅓ of calls.

This precedent can be taken advantage of for VoIP clients where, in the absence of a civic address which can be displayed to the PSAP operator, a geodetic location—just as is used for phase 2 cellular location—is provided in an embodiment of this invention.

However, this does not mean that emergency calls from IP based voice networks need always be restricted to geodetic based location reporting. As discussed, the ESP signaling parameters as defined by NENA includes a parameter called "location description". The NENA specification defines a number of different XML tag based fields that can be used to constitute this parameter. This opens the possibility that the LGS, in responding to an ESPOSREQ request over the E2 interface, can utilise this parameter to also provide a civic address for the caller.

In cellular systems, this parameter has a nominal use around supporting phase 1 capable PSAPs where the location description provided will generally correspond to a street address identifier for the serving base station in the cellular network. However, this use does not preclude the alternative use in IP based voice networks.

Where VoIP clients have a relatively static location—for example, where the client is a conventional telephone form factor device with a relatively fixed desktop location—then the access network, which provides location to the LGS, may opt to provide a civic address encoding in addition to the geodetic location. A discussion on general location determination and the associated signaling is given below.

A valid question is how the emergency services network can know that it is receiving a civic address for the caller rather than a nominal base station address. This can be discriminated in a number of ways. The key is that the emergency network can be aware that it is interfacing to an IP based voice network rather than a cellular network. Three potential ways to perform this discrimination are:

- The emergency network will generally select the E2 interface that it needs to send a request to on the basis of the ESRK associated with the call. ESRKs tend to be allocated to network operators in pools. This same association can allow the emergency network to infer the nature of the connecting network.
- The esposreq response contains a parameter which is the Company ID. This can be used by the emergency network to distinguish IP vs cellular carriers.
- The position data parameter in the esposreq which contains the geodetic location also contains a sub-parameter called "position source" which indicates the technology used to establish the location.
- New code points can be allocated for IP network positioning technologies. This could be used by the emergency network to establish that the location is being provided by an IP voice network.

The example mechanisms discussed above identify how the existing cellular E911 phase 2 infrastructure and interfaces in the emergency network can be effectively reused with little or no modification to support the delivery of caller location from IP based voice networks.

In order to minimise the need to transform and translate the information related to location, in a preferred embodiment the specifications used for this on the E2 interface are reused within the signaling of the IP network. That is the geodetic location coding defined by NENA in the document referred to above as well as the XML tag encodings defined in "Real Time ALI Exchange Interface Agreement—Issue 6.1", AT&T and Pacific Bell, Mar. 25, 1995 by NENA are also preferred for use between the IP network elements as they are delivered through to the LGS.

End to End—Adding Location Determination

There are numerous approaches to location determination within IP networks and any suitable approach can be used in the present invention. A number of things will affect the type of solution put in place. Amongst these are:

- The nature of the connection used by the client. That is, whether it is a domestic broadband connection, an enterprise IP switch connected client, a wireless client connecting via a campus wireless LAN, etc.
- Legacy circumstances. That is, the extent to whether the clients, access devices, and switches have native support for location delivery versus the need to overlay a solution for location determination on existing infrastructure.
- The type of location information and accuracy required for a given target environment. For example, are static

civic addresses with sufficient geodetic accuracy for routing sufficient or is a more accurate geodetic location required in the absence of a civic address?

The NENA website itself has a number of submissions and proposals around different positioning technologies for IP and any one of these may be adopted in a given access network.

The Location Identification Server—LIS

An embodiment of this invention proposes that an intermediate network entity be defined which provides a uniform query interface to the LGS network element such that it need not be concerned with the nature of the positioning technology used.

The newly identified network element is the Location Identification Server (LIS). This network element sits between the LGS and the access network and invokes the applicable positioning technologies. It supports a simple request/response message that allows the LGS obtain the location of a client.

Client Identifier Options

In order to do this, the LGS needs to provide a client identifier which is meaningful to the LIS and significant within the access network that the client is attached through. Types of potential client identifier vary but some candidates are:

- Ethernet MAC address
- MSISDN—international encoding of corresponding dialable digits
- RFC 2486 Network Address Indicator—user@realm style address
- SIP URL address
- Some other network element, e.g. LIS, generated handle to the client that is independent of other addressing schemes.

The above list is by no means definitive but the definition of the query messaging between the LIS and the LGS is defined such that these and other forms of client identification can be supported over this interface. An important driver of the form of client identification supported is which identifier can be provided by the call server function in its request to the LGS. Any practical network deployment will need to ensure that the same client identifier form can be used meaningfully by the call server, LGS, and LIS.

By way of example, in initial implementations of this architecture where the access network and client devices are largely legacy, and without native location determination capabilities, the likely candidate for many deployments may be the MAC address.

An example of an end to end solution using a LIS that employs SNMP bridge MIB polling and MAC address association is described below with reference to FIG. **7**.

Geodetic vs Civic Address Location—Revisited

As discussed above, location may be provided as a geodetic location for the purposes of call routing plus, optionally, a civic address that can be displayed to the PSAP operator. The parameter in the response message from the LIS to the LGS that specifies the returned location preferably supports a coding that supports both of the location formats concurrently. The geodetic location is provided in order to support emergency call routing. Also as discussed above, it is preferred that the specifications used for coding location are the same as those on the E2 interface. That is the geodetic location coding as well as the XML tag encodings defined by NENA are preferably used to encode the location provided to the LGS by the LIS. This eliminates the need to translate and transform this information as it is passed from the LGS to the emergency services network.

The architecture that has been described herein—from the LIS through the LGS, call server, and PVG network entities interfacing to the emergency services network ISUP and E2 interfaces—should meet the needs of emergency calling from VoIP networks well into the future. Further, as more standardisation occurs at the IP access and native positioning support is deployed, this transition to more reliable and accurate location determination will be able to occur seamlessly without impacting the VoIP to emergency network interface. The changes will be perceived as an improvement in coverage and quality of service for VoIP emergency callers as well as ease of deployment for VoIP operators but without impacting the operation of the emergency network generally.

In addition to the above, as the emergency network infrastructure evolves away from the current legacy of CAMA trunks and PAM interfaces, individual PSAPs will be able to interface directly to the IP network. The same functions of call routing and location delivery will still be needed and the mechanisms described can still be utilized. Instead of routing out to ISUP trunks, the call server can direct the call to a direct VoIP based ACD function. The ESP messaging referred to above is already IP based and the option becomes available for updated PSAPs to query the LGS directly instead of their requests being proxied through an ALI.

Using DHCP to Improve Client Integration

Since the identity, location, and capabilities of the LIS will vary from access network to access network, it is preferred that in some embodiments DHCP be used to advise IP clients of the identity of the serving LIS. This permits two major optimisations:

- The client will be able to explicitly register with the LIS so that it is known to that entity for purposes of location. This will also establish a signaling relationship that can be used for advanced positioning mechanisms if supported. It also offers the opportunity for the LIS to assign a client-specific identifier which the client can provide to network services such that no other client key is required for the purposes of location requests through the LGS/LIS network.
- The call setup signaling to the call server can be modified to support the ability of the client to forward the serving LIS identity to the call server. This in turn can be communicated as part of the location request to the LGS, permitting the LGS to have explicit knowledge of the appropriate LIS to query.

These embodiments are illustrated in FIG. **5**.

Supporting International Emergency Calling

It is an interesting characteristic of VoIP networks that the distance between a client user and the call server handling the call processing may be arbitrarily great. A VoIP client can typically use the same call server regardless of the point of attachment to the network. So, the client may be in a different city, a different state, or even a different country.

It has been an implicit assumption in the discussion to date, that the call server has inbuilt knowledge of the LGS that it should inform of the incidence of an emergency call and request routing information from. While this may hold true of a nationwide carrier with points of presence across many states, it may prove difficult for some VoIP network operators to provide the same ubiquity of presence. When

the question of supporting international calling is raised, then it becomes even less likely that this assumption will apply.

This constraint will likely continue for the short term. However, the use of DHCP may, in the future, also provide a mechanism for dealing with this. In this instance, the registration of a client on a local network involves not only an indication of the serving LIS identity but also an indication of the applicable emergency LGS (eLGS).

With this facility, the client can provide the eLGS identity to the call server. This introduces the possibility of a network of regional LGS platforms to serve the VoIP network. The ESRK allocation pools can be efficiently distributed between these LGS and they can retain the responsibility of maintaining the spatial boundary information for the emergency service (PSAP) zones in their regions.

The signaling associated with this scenario is also shown in FIG. **5**. Note that the call server was able to refer to an eLGS in the visited network rather than the one in the subscriber's home network. This allowed the appropriate ESRK for the PSAP in the visited network operator's region to be allocated by that operator. Further, the PSAP in that region only needs to have an E2 interface association with that network's LGS and not the home network LGS.

The arrow labeled in FIG. **5** shows that the DHCP server provides LIS and eLGS identities to the client on initialization. Arrow **1***a* represents the optional step whereby the client registers with the LIS to establish a signaling relationship for future positioning. As shown by arrow **2** the client then provides eLGS and LIS identities to call server on emergency call initiation. Then in the step shown by arrow **3** the call server provides LIS identity to eLGS in emergency call request.

Enterprise Versus Carrier VoIP Network Deployment

In the embodiments described so far it has been an assumption that the VoIP network operator has sufficient points of presence in each of the regions of interest to be able to route the emergency calls onto the local network and into the emergency services network. This is typically true of a public carrier network which operates its own PVG platforms that tandem directly into the public wireline network but it is less likely for an enterprise operating a VoIP network over its intranet.

In the case of an enterprise VoIP operator, this may not be an issue where the PABX or other PSTN gateway utilised by that enterprise is colocated with its user population. However, if the user population is widely geographically distributed via a wide area intranet and/or VPN links and they share a common PSTN gateway, then there is no native mechanism to support routing to the correct PSAP.

- For a colocated user population, the class **5** switch in the local operator network which provides the enterprise service looks after the subsequent routing of the 911 call to the correct selective router and PSAP.
- This local exchange interface does not support the use of an ESRK in the call setup signaling to indicate a preferred route and a local exchange will not tend to support the necessary trunking to remote selective routers for out-of-region callers.
- For small and medium enterprises, it would not necessarily be economical to operate an LGS nor would it be optimal to distribute ESRK pools around arbitrary numbers of enterprises.

Despite these constraints, it is still desirable to utilise the embodiments that have been described herein as the challenge of routing calls from geographically distributed callers

needs to be addressed. While there are alternative proposals these tend to rely on direct dialing local access numbers for PSAPs. While this is effective in the short term, it is by definition bypassing the existing mechanisms and processes for emergency call distribution.

At least two possible approaches to supporting the enterprise environment in the long term exist.

- Through the standards process, the local operator switch interface could be modified such that the ESRK can be delivered in the call setup.

This approach has a number of limitations including the fact that the time lag in defining this signaling and having switch vendors implement and deploy it can be very large. More significantly, it doesn't address the concern that the local operator and switch is unlikely to maintain direct trunks to all required destination selective routers.

- Enterprises can seek emergency service support from public network carriers that support VoIP deployments. This means utilising the LGS and PVG resources of the public carrier but only for the purposes of emergency call routing.

In this situation, the enterprise would still provide the LIS functionality within their intranet IP access. Using the equivalent of the DHCP mechanism described above, the enterprise client can be advised of the carrier LGS applicable to emergency calls in that location and relay this to the call server at call setup. At the same time the identity of the serving LIS can also be relayed via the call server to the LGS.

This arrangement is illustrated in FIG. **6**.

FIG. **7** shows a simplified example of a VoIP deployment where the network operator is a carrier **70** and the subscriber population are within enterprise managed networks **71**, **72**. That is, this shows virtual private voice network deployment, where the call services are operated on an IP network with call serving functionality outsourced from the enterprise to the carrier.

In this example, it is assumed that each of the enterprises operates the voice network under the constraint that all voice clients need to be connected via specific IP switches supporting a standard SNMP bridge MIB 73 that permits port scanning to occur and also permits the MAC address of connected clients to be retrieved.

Further, the client implementation and protocol are conventional but include the delivery of the client MAC address as part of the native call signaling with the call server.

These constraints permit the operation of the network such that the MAC address can be used as a query key between the call server and LGS (Lv) interface and the LGS and LIS (Li) interface. The LIS implementation in this case involves the continuous SNMP polling of managed switches according to provisioned data which includes the list of managed switches, their ports, and the nominal location of the end-cabling attached to those ports—as both a geodetic location and, optionally, a civic address. On each poll cycle, the LIS stores any connected MAC address values against the port records within this wire map.

A query to the LIS from the LGS, then, simply results in the stored location information in this wire map being keyed from the provided client MAC address in the query. This location information is returned for subsequent processing by the LGS as described herein.

This example illustrates how the complexities of location determination in the access network are abstracted away from the rest of the emergency call handling. Other examples of LIS implementations would be those that could map a DSLAM port to a physical home address location for

ADSL broadband internet based subscribers. Again, the details of how this particular LIS performed this function would be hidden from the rest of the VoIP network.

This embodiment provides the advantage that there is now a seamless migration path to native positioning systems that will not impact the network beyond the access interface to the LIS.

Call Back Number Considerations

One of the current limitations of the existing emergency services network is the ability to support callback number reporting to the PSAP where that callback number exceeds the number of digits used for a normal local dialable number. Examples of callback numbers that may not be supported are:

International callback numbers such as international roaming cellular callers or, in future, international roaming VoIP callers.

Enterprise callers to emergency services where the terminal callback number is not delivered in the call setup information.

The use of E2 as a dynamic query interface also facilitates the delivery of callback information. Since this information is delivered out of band from the call setup, it isn't subject to the same constraints as imposed by the selective router and CAMA trunk infrastructure.

The callback number is one of the parameters in the esposreq message in ESP. This allows the originating voice network which uses the E2 interface the ability to deliver an appropriate callback number, if available, for the particular call in progress. The LGS then can also be used to query the access network or be informed by the Call Server, as appropriate, of a callback number to cache in anticipation of the PSAP query.

FIG. **8** is a message sequence chart showing a consolidated end to end signaling flow for several of the embodiments described herein. It includes the scenario of a mid-call location update request from the PSAP.

The invention claimed is:

1. A method of providing a routing key for routing an emergency call from a packet-based communications network node to an emergency services network node in a switched telephone network, the packet-based communications network comprising an access part and a core part, said method comprising the steps, performed at a node of the core part of the packet-based communications network, of:

(i) requesting, during the emergency call, a node in the access part of the packet-based communications network to determine a current geographical location of a terminal from which the emergency call originates;

(ii) receiving information about the determined current geographical location from the node in the access part of the packet-based communications network;

(iii) generating a routing key on the basis of the received information and pre-specified information about geographical locations served by particular emergency service network nodes; and

(iv) storing said generated routing key together with the received information at the node in the core part of the packet-based communications network.

2. A method as claimed in claim **1** wherein said step (ii) further comprises receiving a call-back number from which the emergency call originates.

3. A method as claimed in claim **1** which further comprises providing the stored information to the switched telephone network for receipt by a public safety answering point (PSAP).

4. A method as claimed in claim **3** wherein said stored information is provided via an E2 interface.

5. A computer program stored on a computer readable medium and arranged to control a location gateway server in a packet-based communications network in order to carry out the method of claim **1**.

6. A method of routing an incoming emergency call in a packet-based communications network to an appropriate emergency services answering point in a switched telephone network, the packet-based communications network comprising an access part and a core part, said method comprising:

(I) at a packet-based call server in the core part of the packet-based communications network, receiving the emergency call;

(ii) requesting a node in the access part of the packet-based communications network to determine a current geographical location of a terminal from which the emergency call originates;

(iii) at a location gateway server in the core part of the packet-based communications network, receiving information about the determined current geographical location from the node in the access part of the packet-based communications network and using that to generate a routing key;

(iv) at the call server, routing the emergency call using the generated routing key; and

(v) storing said generated routing key together with the received information at the location gateway server.

7. A method as claimed in claim **6** wherein said information is received at the location gateway server as a result of polling a location information server in the access part of the packet-based communications network.

8. A method as claimed in claim **6** wherein said packet-based communications network is a voice over internet protocol network.

9. A method as claimed in claim **6** wherein said terminal is a nomadic entity.

10. A method as claimed in claim **6** which further comprises making the stored information accessible to an automatic location identification node in an emergency services network.

* * * * *

EXHIBIT F

US006823370B1

(12) **United States Patent**
Kredo et al.

(10) **Patent No.: US 6,823,370 B1**
(45) **Date of Patent: Nov. 23, 2004**

(54) **SYSTEM AND METHOD FOR RETRIEVING SELECT WEB CONTENT**

(75) Inventors: **Thomas J. Kredo**, Rochester, NY (US); **Kenneth J. Kohl**, Penfield, NY (US); **Stephen Knight**, Rochester, NY (US)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 197 days.

(21) Appl. No.: **10/191,081**

(22) Filed: **Jul. 10, 2002**

Related U.S. Application Data

(63) Continuation of application No. 09/420,154, filed on Oct. 18, 1999, now abandoned.

(51) Int. Cl.[7] **G06F 15/16**

(52) U.S. Cl. **709/206**; 709/217; 709/218; 709/219; 709/246

(58) Field of Search 709/204–207, 709/217–219, 245–246; 704/270, 270.1, 277; 379/88.13, 88.14, 88.17, 90.01, 93.14

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,727,159 A 3/1998 Kikinis 709/246
5,761,280 A 6/1998 Noonen et al. 379/93.27
5,799,063 A * 8/1998 Krane 709/219
5,838,682 A 11/1998 Dekelbaum et al. 370/401
5,873,077 A 2/1999 Kanoh et al. 707/3
5,875,436 A 2/1999 Kikinis 705/34
5,884,262 A 3/1999 Wise et al. 709/219
5,926,789 A * 7/1999 Barbara et al. 704/270.1
6,240,448 B1 * 5/2001 Imielinski et al. 709/218
6,298,129 B1 * 10/2001 Culver et al. 379/202.01
6,366,650 B1 * 4/2002 Rhie et al. 709/219
6,621,502 B1 * 9/2003 Nair et al. 709/206
6,643,621 B1 * 11/2003 Dodrill et al. 709/219

* cited by examiner

Primary Examiner—Bharat Barot
(74) *Attorney, Agent, or Firm*—Hunton & Williams LLP

(57) **ABSTRACT**

An operator assisted system helps a caller browse the World Wide Web without requiring the caller use a computer. The invention enables a caller on a POTS or cellular/wireless telephone connection to connect with an operator to search and select Web-based content. The invention collects desired search results and passes them to an IVR (Interactive Voice Response) system for presentation to the POTS/Cellular caller. The invention facilitates high-quality search request interpretation and highly-efficient Web searches by a trained operator, using a visual PC based browser.

12 Claims, 2 Drawing Sheets

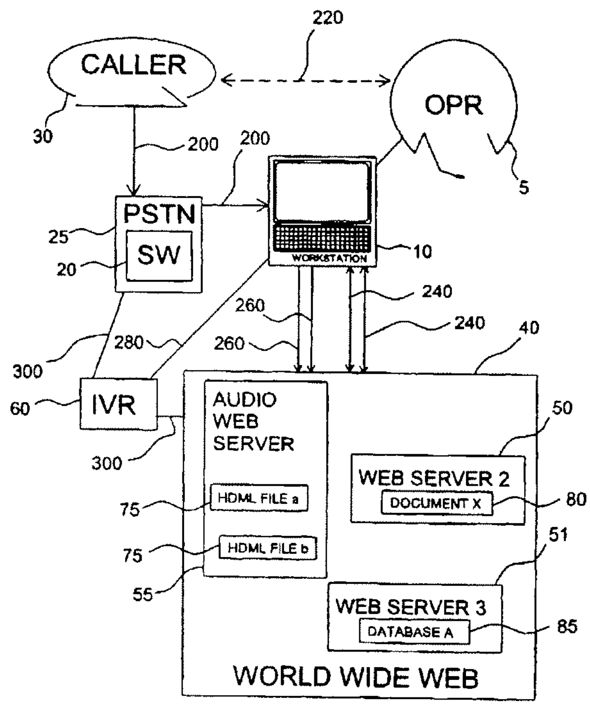


FIG. 1

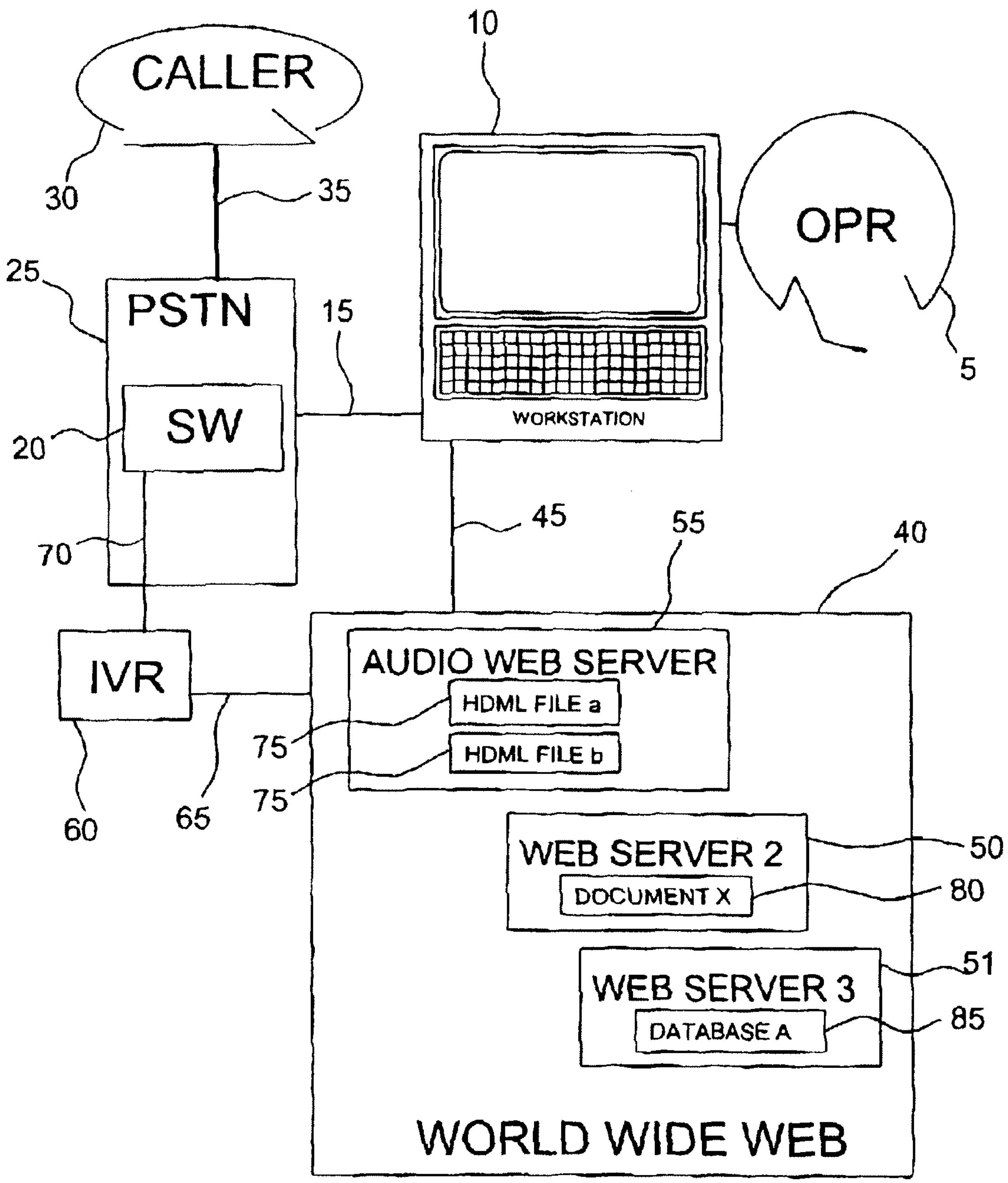
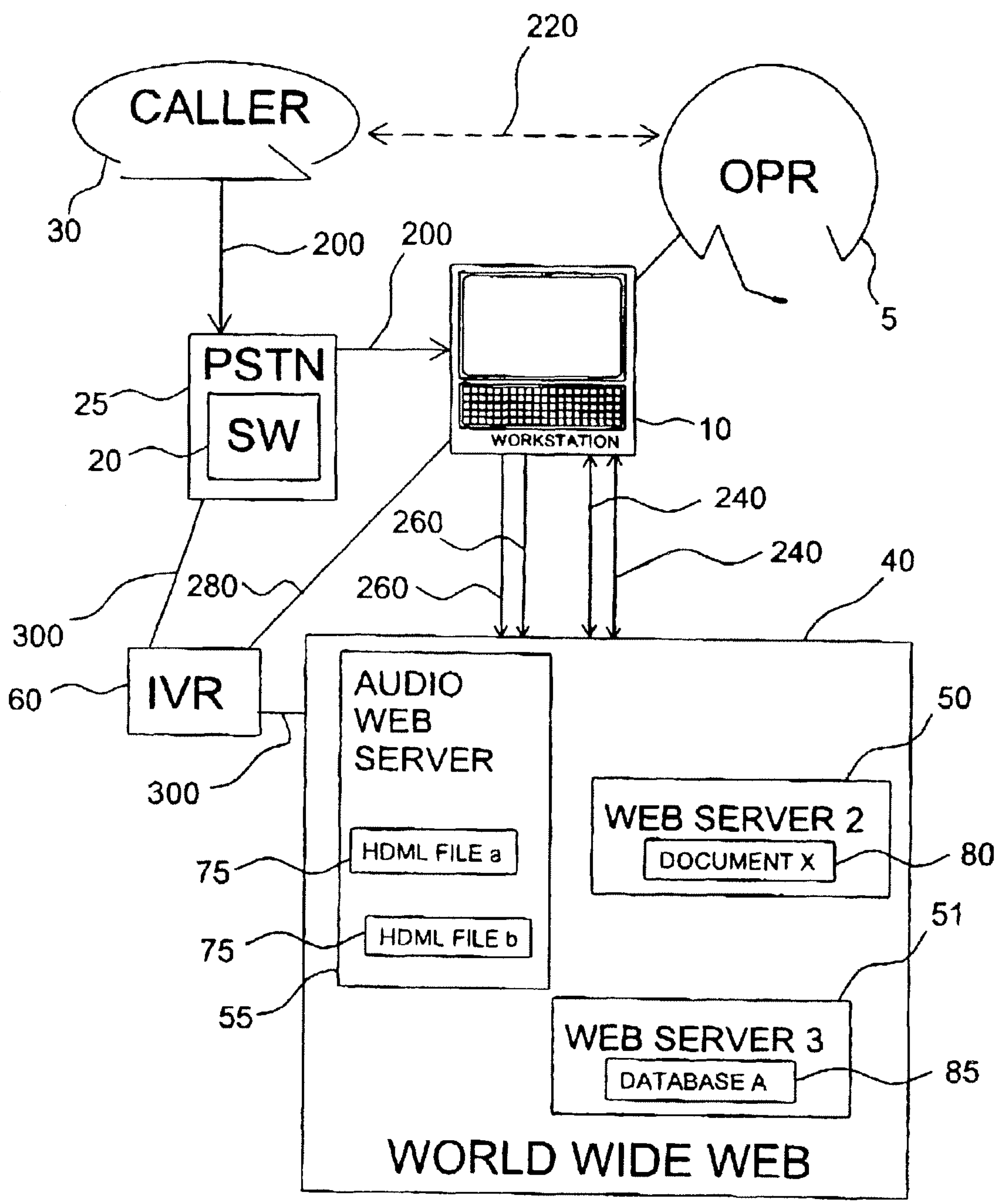

CALLER
10
30
35
OPR
25
PSTN
15
SW
WORKSTATION
5
20
70
45
55
40
IVR
AUDIO WEB SERVER
HDML FILE a
HDML FILE b
75
75
60
65
WEB SERVER 2
DOCUMENT X
50
80
WEB SERVER 3
DATABASE A
51
85
WORLD WIDE WEB

FIG. 2


220
CALLER
OPR
30
200
200
5
PSTN
SW
25
20
10
WORKSTATION
240
260
240
40
280
260
300
IVR
60
AUDIO
WEB
SERVER
50
300
WEB SERVER 2
DOCUMENT X
80
75
HDML FILE a
51
75
HDML FILE b
55
WEB SERVER 3
DATABASE A
85
WORLD WIDE WEB

SYSTEM AND METHOD FOR RETRIEVING SELECT WEB CONTENT

This application is a continuation application of U.S. patent application Ser. No. 09/420,154, filed on Oct. 18, 1999.

BACKGROUND OF INVENTION

This invention relates generally to the retrieval of information from accessible sources on or via the World Wide Web (WWW, or Web). More specifically, the invention relates to the retrieval of Web-based information using ordinary telephones to browse and select the desired information from any of a large number of differently-organized sources available through the Web.

1. Definitions

HDML: Handheld Device Markup Language, and the Wireless Markup Language (WML)—are languages, similar to HTML, that allows the text portion of a Web page to be presented on cellular phones and personal digital assistants (PDAs) via wireless access.

2. Discussion of Prior Art

U.S. Pat. No. 5,884,262 (Wise et al.) describes a telephone voice browser enabling a caller to access Web pages and data including audio files and documents in various formats, browsing the Web and selecting content through the use of speech-to-text analysis of spoken commands and DTMF signals issuing from the caller's telephone connection, and obtaining content through audio and text-to-speech processing onto the caller's connection. This speech-to-text method of browsing and selecting content has three disadvantages.

First, it relies on speech recognition technology. Even after decades of effort, large-vocabulary, speaker-independent, continuous-speech voice recognition does not produce highly-reliable speech-to-text results without unacceptable cost in time and hardware. Regional and ethnic accents and cadences, widely-varying speech habits, unreliable telephone connections, obscure vocabulary and syntax usage, all contribute to increased error rates in automated speech recognition processing. Reduction of error rate requires restriction of one or more of the dimensions of vocabulary range, speech continuity, and speaker-to-speaker variation. Such restriction limits the range of usefulness of the Wise proposal.

Second, for data retrieval, the Wise proposal requires an easy-to-use yet powerful interface between the ordinary telephone user and any number of Web-accessible databases of widely-varying complexity and sophistication. Such a requirement limits the range of users to the few who are satisfied with the results of simple queries and the few who possess sufficient database-search skills to use successfully the results returned by more-complex requests.

Despite progress in 'intelligent' software, complexities of language still limit the range and power of such methods. Much effort has been expended to make database query languages such as SQL more 'user-friendly', but such efforts have forced a tradeoff between usefulness of search results and simplicity of performing the searches. No simple substitute has yet been found for skilled human search practice.

Third, the speech recognition process in Wise must be applied to speech as delivered from a user across an ordinary PSTN POTS line. Speech delivered this way sharply attenuates speech frequencies outside the range of 300–3300 Hz, making its analysis for content significantly more difficult and inconclusive than such analysis for speech delivered with full fidelity. The result is a significantly-increased error rate in recognition, which diminishes the value of Wise for an ordinary user.

Problems inherent in speech recognition technology are compounded by typical low-fidelity telephone connections. Given such problems, the use of voice menus becomes predominant as a way of reducing error rates in the user-browser dialog. Voice menus are time-consuming (particularly where URLs are presented to the user) and limited to a short range of choices, again reducing the value of inventions, such as Wise, using them.

U.S. Pat. No. 5,873,077 (Kanoh et al.) describes a fax-based Web-access method and apparatus allowing a user to exchange faxes with a Website; the Website scans the user's faxes to select a course of action or a collection of data to return via a fax. Kanoh does not describe any voice access or any non-fax telephone usage.

U.S. Pat. No. 5,838,682 (Dekelbaum et al.) describes a dual-link system, using both a network connection and a PSTN line, to establish and use a secure connection for customer-merchant transactions. It does not address customer-driven browsing and searching the Web, independently of a merchant or sales entity.

U.S. Pat. No. 5,761,280 (Noonen et al.) describes a method and interface for Web browsing using telephone DTMF inputs, a menu system, and a display system attached to the telephone. It does not address the use of audio speech-to-text or text-to-speech.

U.S. Pat. No. 5,727,159 Kikinis) describes a system whereby low-end computers similar to personal digital assistants (PDAs) not ordinarily capable of Web browsing may be used to browse the Web. It does not address any telephone interfaces.

U.S. Pat. No. 5,875,436 (Kikinis) describes a work-order transcription and communication system using the Internet. It does not address Web browsing at all.

The Web-On-Call™ product made by General Magic, Inc., is a software product designed to be installed in a Web server, which provides a client user the ability to browse the Web using a telephone, in a manner similar to the Wise proposal. Like the Wise proposal, it relies on automated methods to respond to the user's requests. The Web-On-Call™ product is subject to the same limitations outlined above for Wise.

DESCRIPTION OF THE DRAWINGS

FIG. **1** shows the components and structure of the invention and its supporting telecommunications environment.

FIG. **2** shows the operation of the invention as it takes place within and among the components and structure portrayed in FIG. **1**.

SUMMARY

The invention enables a caller on a POTS or cellular/wireless telephone connection to connect with an operator to search and select Web-based content. The invention collects desired search results and passes them to an IVR (Interactive Voice Response) system for presentation to the POTS/Cellular caller. The invention facilitates high-quality search request interpretation and highly-efficient Web searches by a trained operator, using a visual PC-BASED browser, while utilizing the existing efficiencies of automated announcement IVR systems.

The invention provides a caller with an operator assisted search service that translates Web text to audio so that the caller can intelligently browse the World Wide Web without the caller operating computer. The elements of the system include a workstation, an interactive voice response (IVR) module, an audio web server, and a switch. The workstation and the IVR are connected to the caller via the switch. Both are also connected to an audio web server. The workstation has a browser for searching the Web. When the operator finds the Web information requested by the caller, the operator uses a software program to highlight the information. The software program automatically removes non-textual material, such as graphics. The text information is then stored on an audio web server. The workstation has caller identification information, such as the calling card number of the caller, the CLID of the caller or an access code required for the service. The selected files are stored on the audio web browser in accordance with the identification data of the caller. The workstation hands the call off to the IVR which delivers the selected information to the caller as part of a customary IVR session.

DETAILED DESCRIPTION OF THE INVENTION

Refer to FIG. 1. In the invention's preferred embodiment, the system's components and connections are as follows. A workstation **10** is coupled to a public switched telephone network (PSTN) **25** via an audio link **15**. The operator **5** has a two-way audio connection with the caller **30** through the audio link **15**. Audio link **15** is a 56-kilobit-per-second channel comprising one of the 24 channels of a T1 connection. A caller **30** connects to PSTN **25** via an ordinary POTS telephone line **35**.

Workstation **10** is also connected to the World Wide Web **40** via an Ethernet connection **45** using TCP/IP. That protocol enables workstation **10** to communicate with Web servers **50**, **51** and audio Web Server **55**. A computer program (not shown) operates on workstation **10** for selecting portions of Web pages and for removing non-textual indicia (graphics, photos, etc.) from the selected portions of the Web pages. Details of the construction and operation of such a program are not necessary and are not disclosed because one skilled in the art may implement such programs in a variety of ways. A caller **30** is connected to the PSTN. Switch **20** on the PSTN connects the caller **30** to the operator **5** via audio link **15** and workstation **10**.

An audio announcement system **60** (also called IVR, or Interactive Voice Response) is connected to World Wide Web **40** via an Ethernet connection **65** using TCP/IP. IVR **60** is also connected to switch **20** via a T1 or VoIP connection **70** using TCP/IP. That protocol enables IVR **60** to communicate with audio Web Server **55** and switch **20**.

Audio Web Server **55** is a computer coupled to the Web **40**. It includes a central processing unit, memory for storing operating system and application programs, and input and output devices for receiving and sending transmissions over the Web **40**. Audio Web Server **55** contains memory storage space adequate to store a number of HDML or WML files **75** that can be played for caller **30**. Web servers **50**, **51** contain documents **80** and databases **80**, **85**, respectively. Each database includes information retrievable by operator **5** using workstation **10**.

Operation of Invention

Refer to FIG. 2. In the invention's preferred embodiment, a caller **30** uses the PSTN **25** to make a call **200** by dialing a special access number such as 311. Switch **20** routes call **200** to an available operator workstation **10** serving operator **5**. A conversation **220** between caller **30** and operator **5** results in a number of Web-based searches **240** by operator **5** through Web documents **80** and databases **85** until the information desired by caller **30** is visible to operator **5**. Operator **5** uses conventional browser software to highlight the information that will be announced to caller **5**. Operator **5** then invokes a software program in workstation **10**, which accepts the highlighted information, removes all graphic images and other non-text material, and performs a transmission **260** of the remaining text, along with the URL (universal record locator) of the source file **80** or **85**, into an HDML (or WML) file **75** on a local Web Server **55**. The invention automatically associates the selected HDML files **75** with a specific caller **30**. The HDML files **75** are stored in a directory of the Audio Web Server **55** under the CLID (Calling Line ID) number for caller **30** as qualifying file identification.

Operator **5** then establishes a connection **280** with IVR **60**, and releases caller **30** to IVR **60** and terminates conversation **200**. IVR **60** retrieves HDML files **75** using the CLID of caller **30** to identify file **75**, and announces HDML files **75**, using well-known Text-to-Speech or WAV file technology, by initiating a dialog **300** with caller **30**. To help caller **30** select and operate on files **75**, IVR **60** presents caller **30** with a voice menu of additional options such as back, forward, stop, fax the original URL page to a phone number, next file, previous file, exit. Using voice menus in a dialog **300** with audio subsystem **60**, caller **30** navigates files **75** with DTMF tones to specify commands to be carried out by audio subsystem **60**. Caller **30** terminates connection **300** by hanging up.

Alternate Embodiments of the Invention

The invention may be used with display telephones such as an ADSI phone. ADSI phones are capable of presenting visual displays of significant desirable text portions of retrieved Web pages. With an ADSI phone a caller may directly display HDML files. In an alternate embodiment, the workstation **10** or IVR **60** may determine from PSTN **200** whether or not the caller is using an ADSI instrument, whether or not the ADSI instrument's display is usable. If it the ADSI display feature is useable, the workstation **10** or IVR **60** sends Web page text directly from the stored HDML files **75** to the phone's display. That embodiment eliminates

the need for text-to-speech translation. The ADSI phone can also display menus and process menu selections for navigation among HDML files. The invention sends menus directly to the ADSI phone's display and processes caller menu selections directly, again eliminating any text-to-speech translation.

While the invention uses HDML or WML formats, no particular format is required. Specific implementations of the invention may employ any format for file storage which is usable in the manner described in the operation of the invention. Likewise, the use of Ethernet and TCP/IP is not required and systems using the invention may employ any link type or link protocols which meet the functional requirements of the invention's operation.

The use of the PSTN itself, with its switching equipment, is not required. Those skilled in the art understand that the invention may employ connection, switching, and voice-over-IP capabilities of the Internet to deliver the same services as described in the preferred embodiment above.

The invention may, without loss of its essential character and workings, employ any workable combination of the choices listed in these alternative embodiments.

Illustrative Example

A caller using the invention requests legal assistance from an operator. The operator determines the type of assistance needed, retrieves a series of listings of law firms meeting the caller's needs, stores the listings in HDML form, and activates the audio announcement subsystem. The caller hears the following announcement, delivered by the audio announcement subsystem:

"Three firms were found which meet the requirements you listed. If you wish to hear a full description for Hanford, Sills & Harvey, press 1. For Shakeman & Torrelli, press 2. For Willis, Snipes, Cruise, Jackson & Fishburne, press 3. If during the announcement for a firm you would like to be connected immediately to that film, press the pound sign . . . "

The caller presses 2. The audio subsystem responds by speaking the Web page text retrieved from the Shakeman and Torrelli Web page. As the announcement ends, the caller presses the pound sign on the phone, and the audio subsystem signals the switch to transfer the call to the Shakeman and Torrelli phone number. (This immediate connection capability is currently a feature of available call-processing subsystems.) The call then proceeds as an ordinary telephone call between the caller and the firm of Shakeman and Torrelli.

The invention could carry the caller further into the web pages of the selected site. For example, the site may include biographies of members of the Shakeman & Torrelli firm. At the request of the caller **30**, the operator highlights one or more of the biographies and stores them on the Audio Web Server **55** for playing by IVR **60**.

Those skilled in the art understand that workstation **10** and audio web server **55** are computers that include central processing units, memories, operating system programs and application programs. The switch **20** is also a computer-controlled switch that connects one caller to another. Likewise, the IVR **60** is computer-controlled equipment for generating audio signals and playing audio files in response to inputs received from callers.

Conclusion, Ramifications, and Scope of Invention

The invention offers a simple, clean and functionally powerful method for a caller using an ordinary telephone and telephone infrastructure to access highly-specific information stored on the World Wide Web without requiring recourse to demandingly sophisticated technology and interfaces on the part of the caller.

The invention takes advantage of already-available operator services to eliminate the need for complex search engines in locating desirable Web data, along with the significant skill requirements such search engines impose on their users.

The invention avoids involving an ordinary caller in the complex requirements and limitations of automated speech-recognition technology, rendering its services more attractive to those challenged by such restrictions on their everyday speech.

The invention, while using the current telephony infrastructure and interfaces in their most common forms, still offers the supplier of its services the ability to upgrade service capabilities to accommodate new technologies such as the ADSI phone and voice-over-IP telephone service. Such upgrades can be performed progressively at limited incremental cost, making them attractive to potential service suppliers.

The invention represents, therefore, a new revenue opportunity for its suppliers by leveraging their expertise in automated Operator Services systems. Current telephony service suppliers would be able to extend their capabilities economically by using their existing infrastructure.

From the above descriptions, figures and narratives, the invention's advantages in supplying a telephone caller with convenient and rapid access to information from the World Wide Web should be clear.

Although the description, operation and illustrative material above contain many specific features, those features should not be construed as limiting the scope of the invention but as merely providing illustrations and examples of some of the preferred embodiments of this invention.

Thus the scope of the invention should be determined by the appended claims and their legal equivalents, rather than by the examples given above.

What is claimed is:

1. A system for providing a caller with audio translations of selected portions of World Wide Web pages comprising:

a workstation, an interactive voice response module, an audio web server, and a switch;

the workstation in communication with the interactive voice response module, the audio web server, and the World Wide Web, said workstation including means for selecting portions of pages from locations on the World Wide Web and storing said selected portions as files in the audio web server;

the audio web server in communication with the workstation, the interactive voice response module and the World Wide Web for storing files selected by the workstation;

the interactive voice response module in communication with the audio web server and the workstation for generating audio signals corresponding to the files stored on the audio web server;

the switch for connecting a telephone caller to the workstation and for connecting the interactive voice response module to the caller.

2. The system of claim 1 wherein the switch generates data signals identifying the caller and the workstation stores the selected Web page portions in the 25 audio web server in accordance with the caller identification data signals.

3. The system of claim 2 wherein the audio web server comprises a computer with a memory for storing selected portions of Web pages in accordance with caller identification data signals.

4. The system of claim 1 wherein the workstation further comprises a software program for selecting portions of Web pages and removing non textual indicia from said selected portions.

5. The system of claim 1 wherein the workstation is administered by a live operator.

6. A method for assisting a caller in browsing the World Wide Web comprising the steps of:

selecting one or more Web pages or portions of Web pages;

removing non-textual indicia from the selected pages or portions of pages to form selected files and storing the selected files on an audio server in accordance with identification data corresponding to the caller;

conducting an interactive voice response session with the caller; and

playing one or more of the selected files to the caller in accordance with the results of the interactive voice response session.

7. The method of claim 6 where the one or more Web pages or portions of Web pages are selected by a live operator.

8. A computer workstation with a network connection to the World Wide Web for selecting and translating portions of World Wide Web pages comprising:

means for receiving and transmitting telephone voice signals over a public switched telephone network or a packet switched data network;

means for browsing the World Wide Web and for selecting pages or portions of pages at locations on the World Wide Web, wherein said selected pages or portions of pages include text and non-text indicia;

means for removing non-text indicia from the selected pages or portions of pages to form a selected text file;

means for storing the selected text file; and

means for converting the stored text file to audio signals representative of the selected text and transmitting the audio signals to a telephone caller.

9. The computer workstation of claim 8 further comprising:

means for initiating a voice inquiry to a caller; and

means for generating an output voice signal corresponding to the stored text file.

10. The computer workstation of claim 8 further comprising:

means for receiving a caller identification signal representative of the identity of a caller; and

means for storing the selected Web page portions in accordance with the caller identification signal.

11. The computer workstation of claim 8 wherein the means for removing non-textual indicia from the selected pages or portions of pages comprises a software program for selecting portions of Web pages and removing non-textual indicia from said selected portions.

12. The system of claim 8 wherein the means for browsing the World Wide Web and for selecting pages or portions of pages at locations on the World Wide Web are operated by a live operator.

* * * * *

EXHIBIT G

US007035390B2

(12) **United States Patent**
Elliott

(10) **Patent No.: US 7,035,390 B2**
(45) **Date of Patent: Apr. 25, 2006**

(54) **USER CONTROLLED CALL ROUTING FOR MULTIPLE TELEPHONY DEVICES**

(75) Inventor: **Stephen Bennett Elliott**, Allen, TX (US)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 225 days.

(21) Appl. No.: **10/384,047**

(22) Filed: **Mar. 7, 2003**

(65) **Prior Publication Data**

US 2004/0174976 A1 Sep. 9, 2004

(51) **Int. Cl.**
H04M 3/42 (2006.01)

(52) **U.S. Cl.** **379/201.02**; 379/201.03; 379/207.02

(58) **Field of Classification Search** 370/352–356; 379/88.18–88.19, 88.2, 88.21, 88.26, 201.02, 379/201.03, 207.12, 207.15, 210.02, 211.01, 379/211.02, 211.03
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,313,035	A *	1/1982	Jordan et al.	379/211.02
5,206,901	A *	4/1993	Harlow et al.	379/211.04
5,243,645	A *	9/1993	Bissell et al.	379/211.02
5,329,578	A *	7/1994	Brennan et al.	379/211.03
5,550,907	A *	8/1996	Carlsen	379/211.02
5,724,411	A *	3/1998	Eisdorfer et al.	379/93.23
5,793,859	A *	8/1998	Matthews	379/211.03
5,802,160	A *	9/1998	Kugell et al.	379/211.04
5,805,587	A *	9/1998	Norris et al.	370/352
5,896,448	A	4/1999	Holt	379/211
5,905,789	A *	5/1999	Will	379/211.03
5,946,386	A	8/1999	Rogers et al.	379/265
5,978,673	A	11/1999	Alperovich et al.	455/417
5,999,611	A *	12/1999	Tatchell et al.	379/211.02
6,011,843	A	1/2000	Hochman et al.	379/211
6,094,478	A	7/2000	Shepherd et al.	379/211
6,104,799	A *	8/2000	Jain et al.	379/211.01
6,125,176	A *	9/2000	Foladare et al.	379/211.02
6,130,938	A *	10/2000	Erb	379/211.02
6,141,556	A	10/2000	Dougherty et al.	455/445
6,144,644	A *	11/2000	Bajzath et al.	370/259
6,208,854	B1	3/2001	Roberts et al.	455/417
6,301,350	B1	10/2001	Henningson et al.	379/220.01

(Continued)

FOREIGN PATENT DOCUMENTS

EP 0484067 5/1992

(Continued)

OTHER PUBLICATIONS

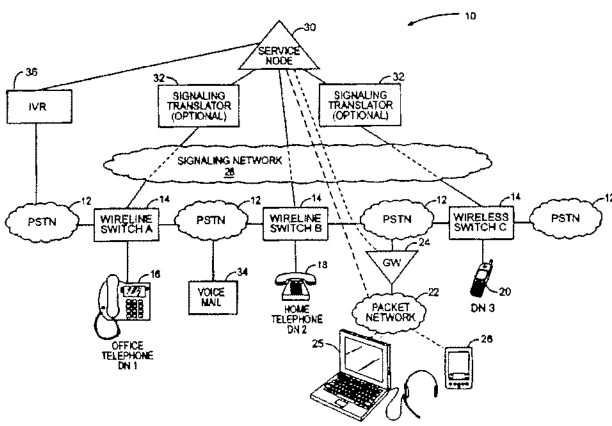
International Search Report for PCT/IB2004/000619 mailed on Sep. 1, 2004.

Primary Examiner—Bing Q. Bui
(74) *Attorney, Agent, or Firm*—Withrow & Terranova, PLLC

(57) **ABSTRACT**

The present invention provides a service node capable of coordinating call processing for incoming calls intended for any one of multiple telephony devices of a given entity, such as a business or individual user. Switching devices, such as traditional telephony switches or internetworked gateways controlling call routing, are configured to interact with the service node to determine how to handle incoming calls to the telephony devices that they serve. Through a variety of techniques, the entity can dynamically instruct the service node how to route incoming calls, and preferably, upon putting those instructions into effect, the service node will send an alert to the entity. The instructions are used by the service node to create call processing rules for the switching devices to apply to an incoming call intended for any of the entity's telephony devices.

44 Claims, 5 Drawing Sheets



U.S. PATENT DOCUMENTS

6,330,322	B1 *	12/2001	Foladare et al.	379/211.01
6,366,661	B1 *	4/2002	Devillier et al.	379/211.01
6,377,668	B1 *	4/2002	Smock et al.	379/142.08
6,678,366	B1	1/2004	Burger et al.	379/211.03

FOREIGN PATENT DOCUMENTS

EP	1 043 902 A2	10/2000
WO	WO 98/35514	8/1998
WO	WO 00/41542	7/2000

* cited by examiner

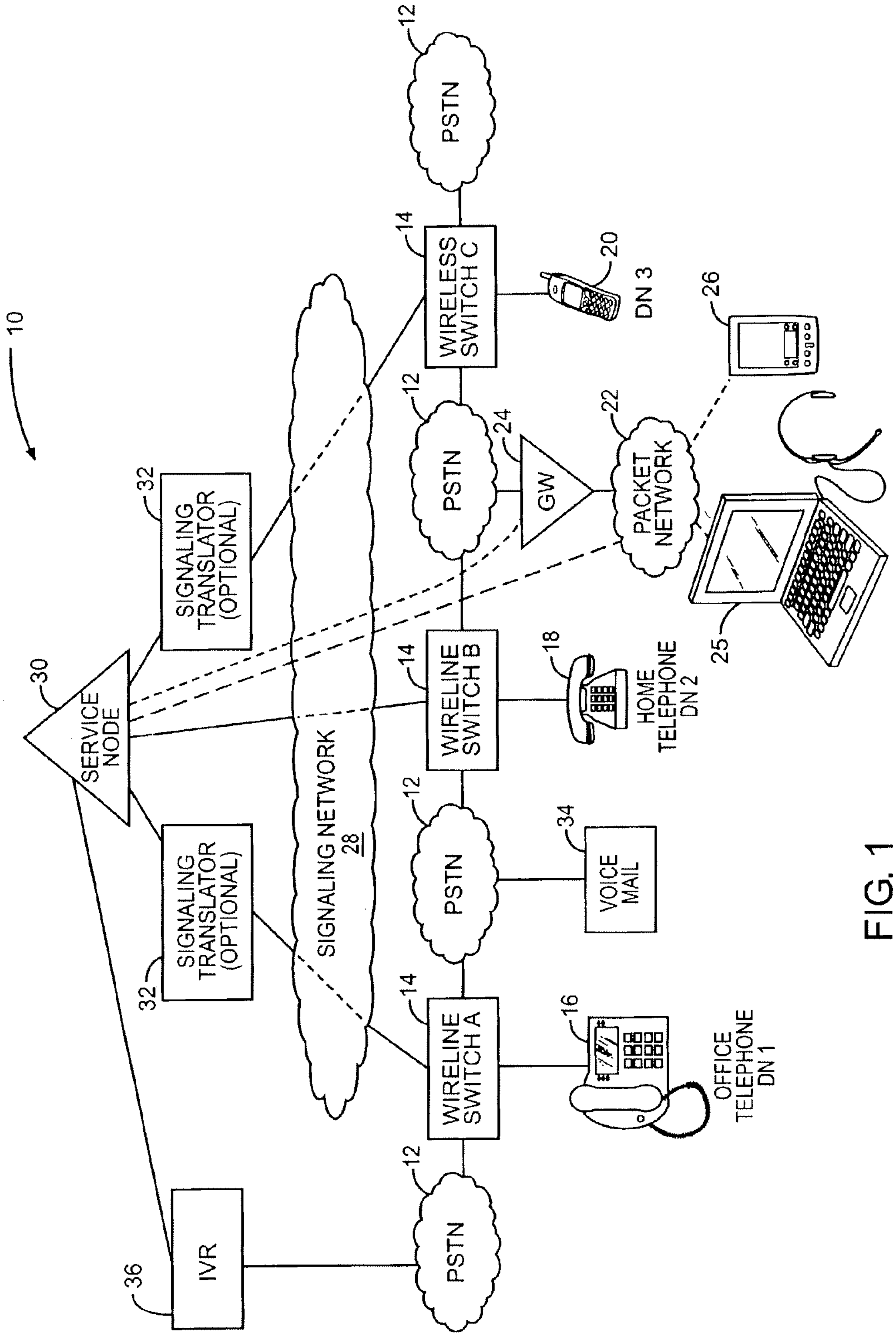
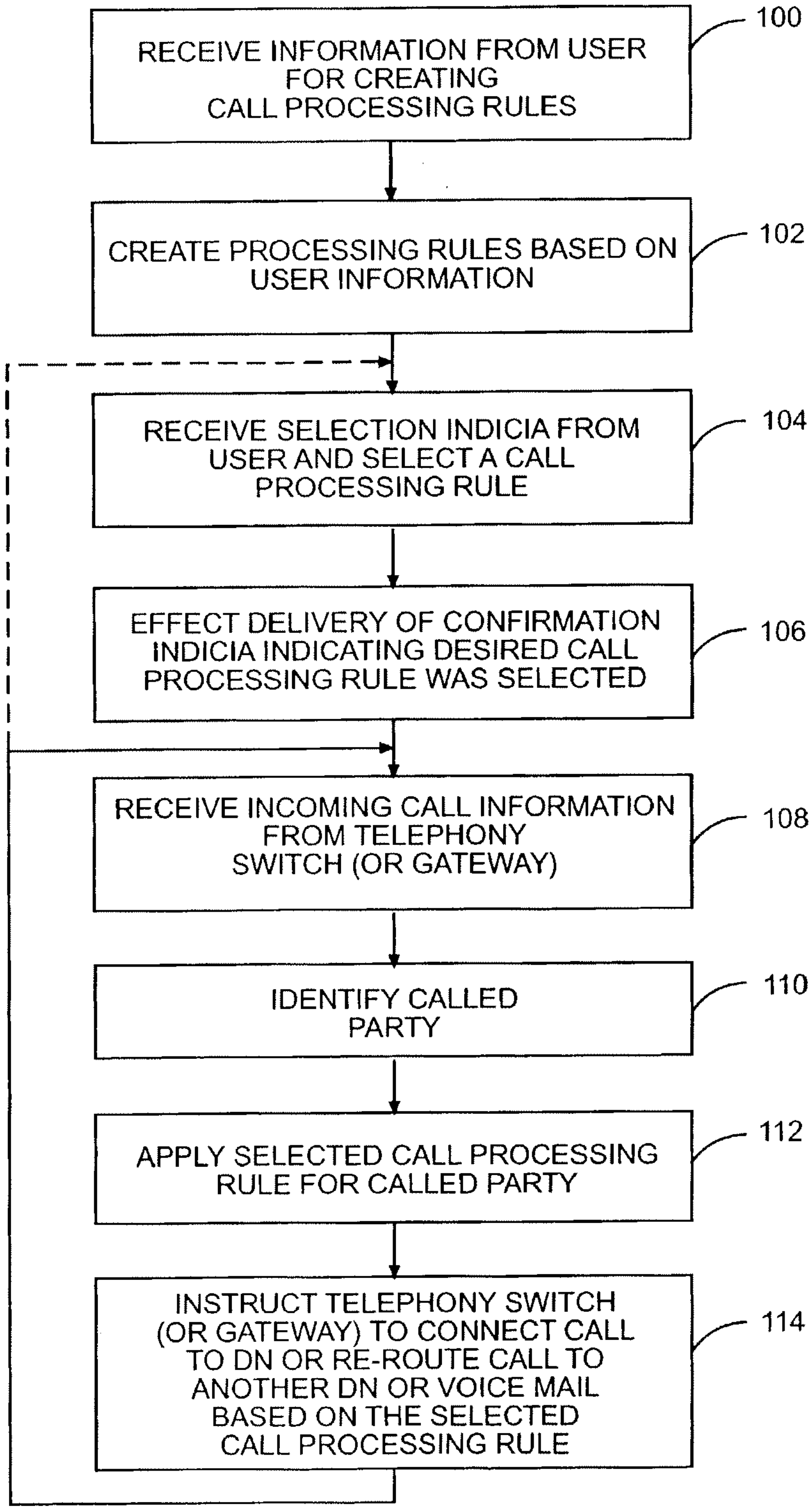

10
SERVICE NODE
30
SIGNALING TRANSLATOR (OPTIONAL)
32
SIGNALING TRANSLATOR (OPTIONAL)
32
IVR
36
SIGNALING NETWORK
28
PSTN
12
WIRELINE SWITCH A
14
WIRELINE SWITCH B
14
WIRELESS SWITCH C
14
GW
24
PACKET NETWORK
22
VOICE MAIL
34
16
OFFICE TELEPHONE DN 1
18
HOME TELEPHONE DN 2
20
DN 3
26
25

FIG. 1



RECEIVE INFORMATION FROM USER FOR CREATING CALL PROCESSING RULES
100
CREATE PROCESSING RULES BASED ON USER INFORMATION
102
RECEIVE SELECTION INDICIA FROM USER AND SELECT A CALL PROCESSING RULE
104
EFFECT DELIVERY OF CONFIRMATION INDICIA INDICATING DESIRED CALL PROCESSING RULE WAS SELECTED
106
RECEIVE INCOMING CALL INFORMATION FROM TELEPHONY SWITCH (OR GATEWAY)
108
IDENTIFY CALLED PARTY
110
APPLY SELECTED CALL PROCESSING RULE FOR CALLED PARTY
112
INSTRUCT TELEPHONY SWITCH (OR GATEWAY) TO CONNECT CALL TO DN OR RE-ROUTE CALL TO ANOTHER DN OR VOICE MAIL BASED ON THE SELECTED CALL PROCESSING RULE
114

FIG. 2

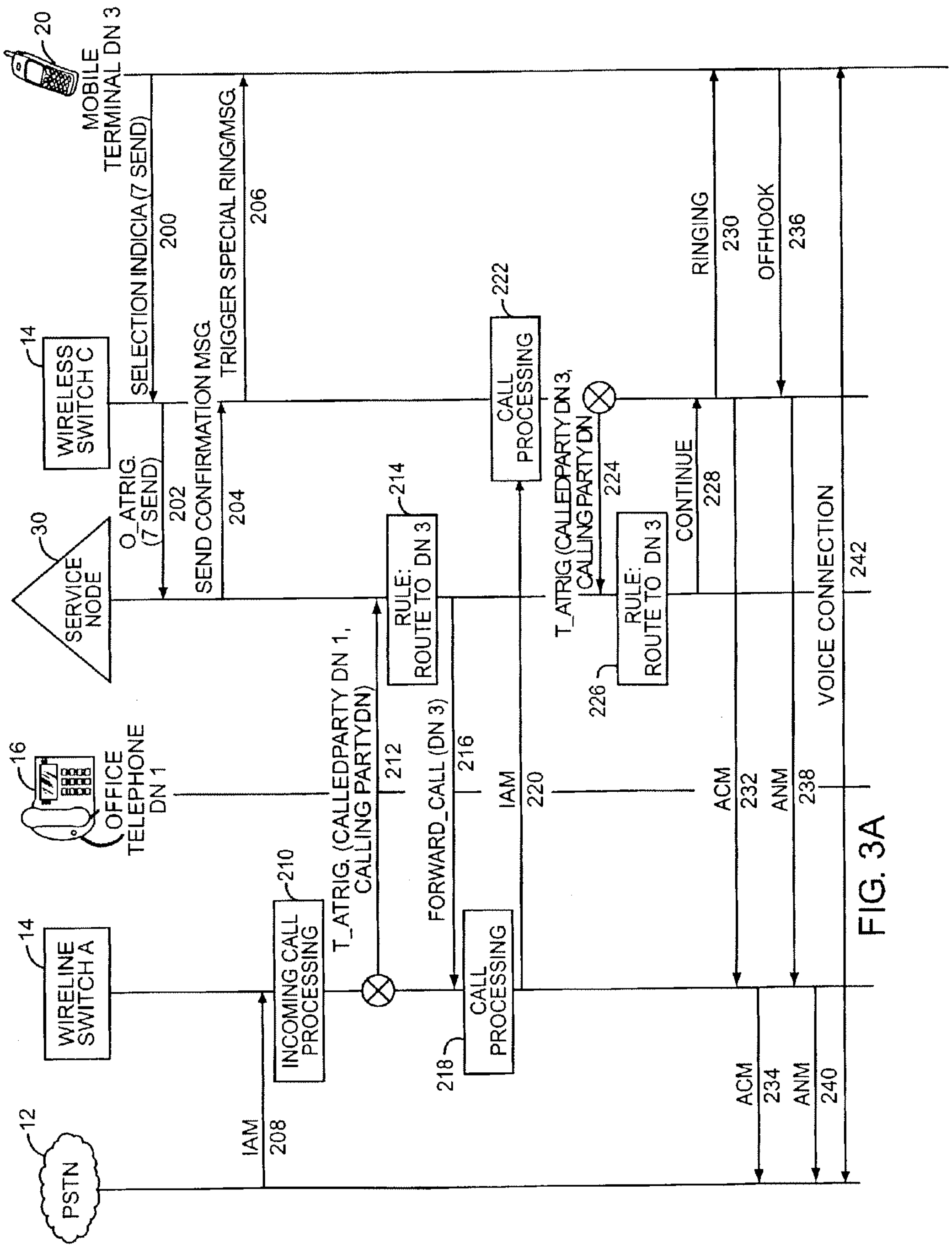
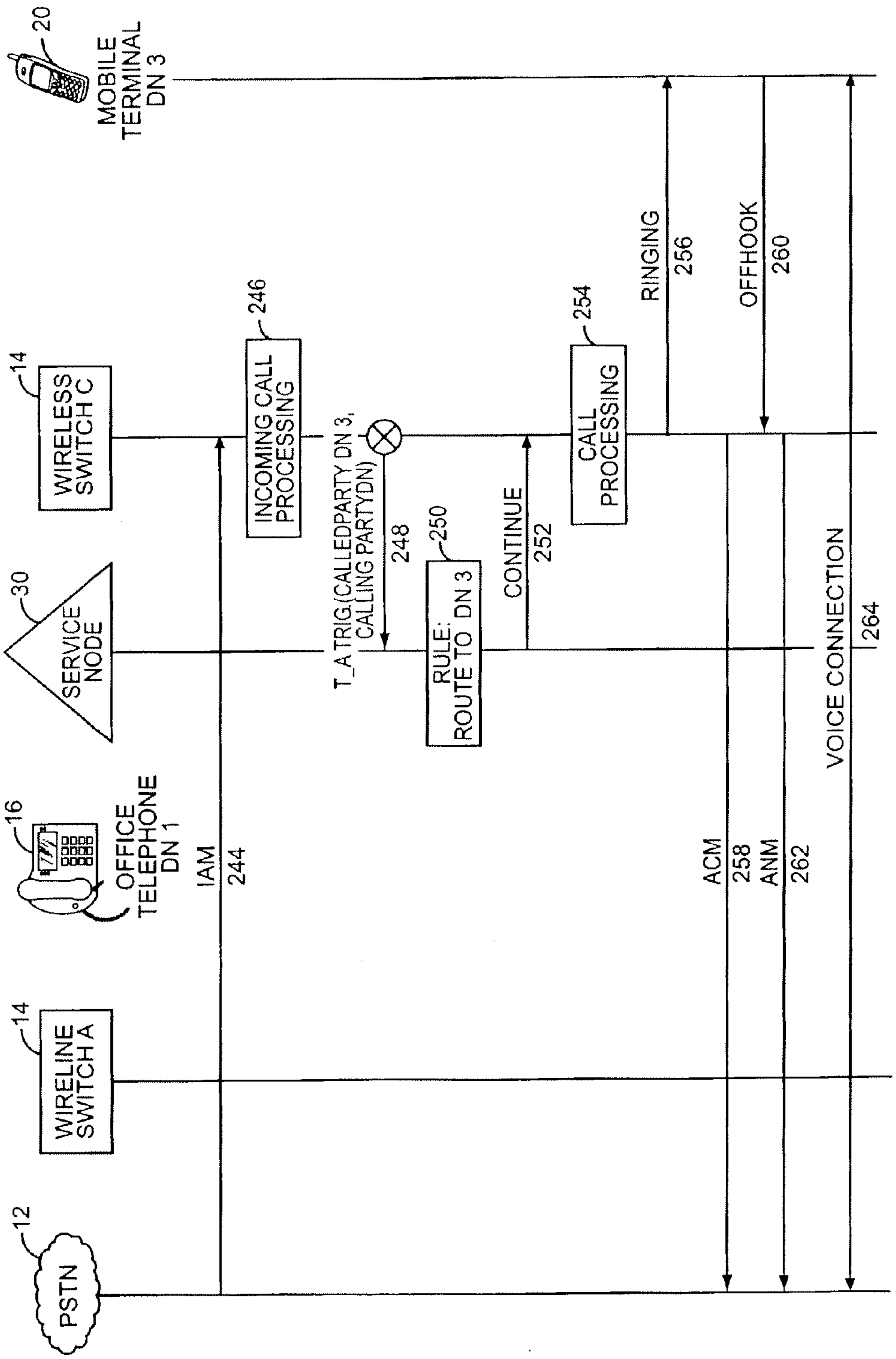

20
MOBILE TERMINAL DN 3
14
WIRELESS SWITCH C
30
SERVICE NODE
16
OFFICE TELEPHONE DN 1
14
WIRELINE SWITCH A
12
PSTN
SELECTION INDICIA (7 SEND)
200
O_ATRIG. (7 SEND)
202
SEND CONFIRMATION MSG.
204
TRIGGER SPECIAL RING/MSG
206
IAM
208
INCOMING CALL PROCESSING
210
T_ATRIG. (CALLEDPARTY DN 1, CALLING PARTYDN)
212
RULE: ROUTE TO DN 3
214
FORWARD_CALL (DN 3)
216
CALL PROCESSING
218
IAM
220
CALL PROCESSING
222
T_ATRIG (CALLEDPARTY DN 3, CALLING PARTY DN)
224
RULE: ROUTE TO DN 3
226
CONTINUE
228
RINGING
230
ACM
232
ACM
234
OFFHOOK
236
ANM
238
ANM
240
VOICE CONNECTION
242

FIG. 3A



MOBILE TERMINAL DN 3
20
WIRELESS SWITCH C
14
SERVICE NODE
30
OFFICE TELEPHONE DN 1
16
WIRELINE SWITCH A
14
PSTN
12
IAM
244
INCOMING CALL PROCESSING
246
T_A TRIG.(CALLEDPARTY DN 3, CALLING PARTYDN)
248
RULE: ROUTE TO DN 3
250
CONTINUE
252
CALL PROCESSING
254
RINGING
256
ACM
258
OFFHOOK
260
ANM
262
VOICE CONNECTION
264

FIG. 3B

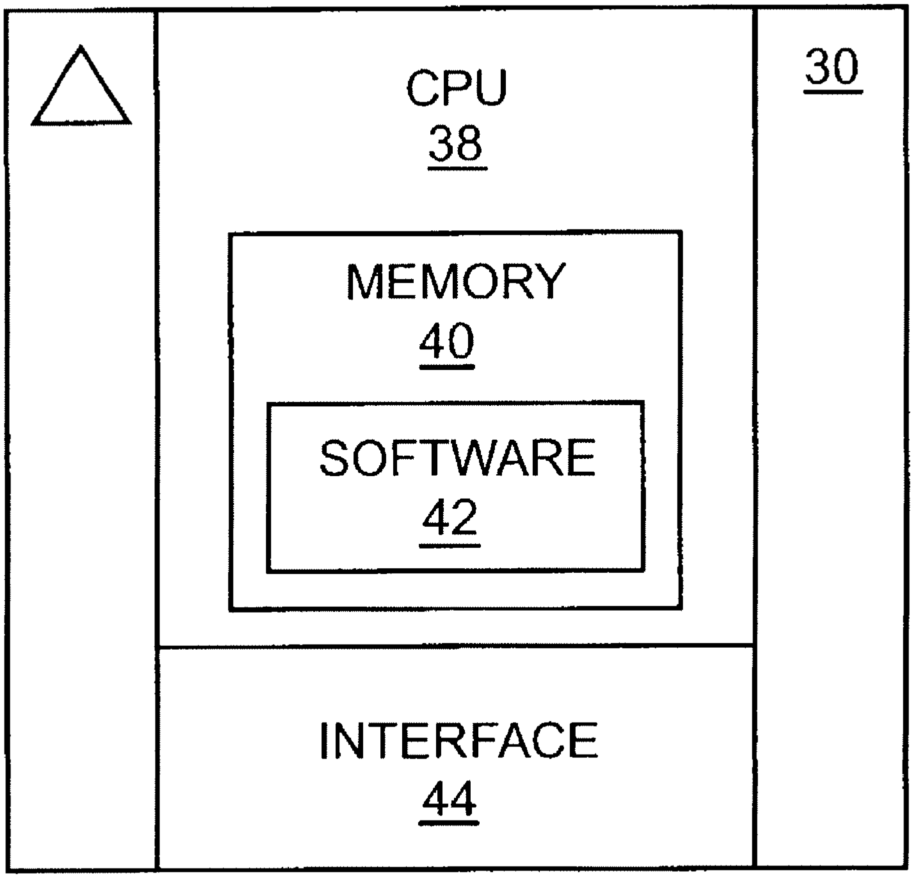


FIG. 4

USER CONTROLLED CALL ROUTING FOR MULTIPLE TELEPHONY DEVICES

FIELD OF THE INVENTION

The present invention relates to call processing, and in particular to allowing a user to control call routing for multiple telephony devices associated with the given user.

BACKGROUND OF THE INVENTION

Telephony users today have multiple communication devices with which to stay in touch with the world. Managing these various communication devices in a consistent and efficient manner is increasingly challenging. Callers trying to contact a user often do not know which directory number should be used to reach the user, and will often have to make multiple call attempts, and in the process, leave multiple voicemails in different voicemail systems of the user. Attempts to minimize these complications have led to "one number" services, where a user can adopt a single directory number for multiple telephony devices. Unfortunately, these services are not widely available, and have proven difficult to implement across different communication technologies and different service providers. Further, the one number approach often forces a user to adopt a new directory number, which is typically undesirable once the original directory numbers have become widely known and used by other parties.

Accordingly, there is a need for a way to allow a user to efficiently and effectively control how incoming calls are routed between multiple telephony devices associated with the user, regardless of the telephony device for which the incoming calls were originally intended.

SUMMARY OF THE INVENTION

The present invention provides a service node capable of coordinating call processing for incoming calls intended for any one of multiple telephony devices of a given entity, such as a business or individual user. Switching devices, such as traditional telephony switches or internetworked gateways controlling call routing, are configured to interact with the service node to determine how to handle incoming calls to the telephony devices that they serve. Through a variety of techniques, the entity can dynamically instruct the service node how to route incoming calls, and preferably, upon putting those instructions into effect, the service node will send an alert to the entity. The instructions are used by the service node to create call processing rules for the switching devices to apply to an incoming call intended for any of the entity's telephony devices to effectively route the calls to the intended telephony device or other ones of the entity's telephony devices, route the call to a desired voicemail system, provide call screening or blocking, and any other desired call control activity dictated by the entity's instructions. Based on the call processing rules selected by the given entity, the service node will instruct the switching devices how to route the incoming calls to any of the telephony devices associated with the entity.

Preferably, the entity provides instructions to the service node directly, indirectly through an interactive voice response system, or via a personal computing device. Direct instructions via the telephony devices may include simply toggling a wireline device off hook and then back on hook, dialing any number or set of numbers followed by a send instruction on a wireless device, or simply selecting an icon

from a telephony control application on the personal computing device, which will send the instructions to the service node via the Internet or like packet-switched network. Packet-switched telephony applications may operate in either of these fashions. For wireline and wireless applications, the associated telephony switches will recognize these signals and be provisioned to send appropriate messages to the service node. The service node will identify the user from the message sent from the switching devices, place select call processing rules in effect based on the instructions, and send a message back to the switching device to trigger a proper alert to the entity.

The alerting methods may take numerous forms. For example, alerting in a wireline application may include providing a special ring signal, lighting a lamp, or providing a stutter dial tone on the wireline telephony device. For a wireless application, the alert may include a special ring signal, or reception of an instant message via a short messaging service, email, or instant messaging service. An alert to the personal computing device may simply be a pop-up acknowledgement message. Typically, the alerts are sent to the devices from which the instructions were initiated, but certain embodiments provide for alerts to one or more of the other telephony devices or personal computing devices regardless of the device from which the instructions were initiated. The entity may be able to access the service node in any number of ways to establish a profile defining the call processing rules to implement based on the instructions provided by the entity.

Those skilled in the art will appreciate the scope of the present invention and realize additional aspects thereof after reading the following detailed description of the preferred embodiments in association with the accompanying drawing figures.

BRIEF DESCRIPTION OF THE DRAWING FIGURES

The accompanying drawing figures incorporated in and forming a part of this specification illustrate several aspects of the invention, and together with the description serve to explain the principles of the invention.

FIG. **1** provides an exemplary communication environment according to one embodiment of the present invention.

FIG. **2** is a flow diagram providing an operational overview of the present invention according to one embodiment.

FIGS. **3**A and **3**B are an exemplary call flow diagram for a first scenario according to one embodiment of the present invention.

FIG. **4** is a block representation of a service node constructed according to one embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The embodiments set forth below represent the necessary information to enable those skilled in the art to practice the invention and illustrate the best mode of practicing the invention. Upon reading the following description in light of the accompanying drawing figures, those skilled in the art will understand the concepts of the invention and will recognize applications of these concepts not particularly addressed herein. It should be understood that these concepts and applications fall within the scope of the disclosure and the accompanying claims.

The present invention allows a user associated with multiple telephony devices having unique directory numbers or addresses to dynamically select rules for controlling the routing of these calls to one of the telephony devices associated with the user, or to a common voicemail system. Regardless of the telephony device being called, the incoming call may be routed, blocked, screened, or otherwise handled according to a defined set of rules automatically based on instructions received by the called party. In particular, the called party will effectively signal a service node from one of the telephony devices or a properly configured personal computing device to provide instructions to the service node for selecting a call processing rule. Based on the call processing rule, all incoming calls to the associated telephony devices will be controlled accordingly. Preferably, the service node will provide an alert to the called party indicative of the instructions being successfully received and the desired rule set implemented.

As an example, a user (called party) may signal the service node from a first telephony device associated with the user, and the signal triggers the service node to direct all incoming calls intended for any of the user's telephony devices to the first telephony device. A call originated by a caller to a second telephony device associated with the user may be automatically rerouted to the first telephony device associated with the user. As described below and as will be recognized by those skilled in the art, the present invention provides tremendous flexibility in allowing the called party to control routing of incoming calls for all of the called party's telephony devices, even those associated with different networks and network technologies. Further detail regarding the operation of the present invention and select examples are provided after an overview of the communication environment capable of implementing the concepts of the present invention.

With reference to FIG. **1**, a communication environment **10** is illustrated as being centered about the public switched telephone network (PSTN) **12**, which is operatively coupled to other networks supporting wireless and packet-based communications. Traditionally, telephony switches, such as wireline switches **14** (A and B) are associated with the PSTN **12** and serve respective telephony devices, such as the office telephone **16**, which is serviced by wireline switch **14**A, and a home telephone **18**, which is serviced by wireline switch **14**B. The office telephone **16** and the home telephone **18** have respective directory numbers (DNs) DN **1** and DN **2**. Calls intended for either of these telephony devices are generally routed via the respective wireline switches **14**A and **14**B.

For wireless communications, a wireless switch **14** (C), such as a mobile switching center (MSC), will support communications with a mobile terminal **20**, such as a wireless telephone or wireless personal digital assistant (PDA). Assume that the mobile terminal **20** is associated with a directory number, DN **3**, and all calls routed to the mobile terminal **20** will be routed through the appropriate wireless switch **14**C.

For packet-based telephony, such as voice over packet (VoP), interaction between a packet network **22** and the PSTN **12** is facilitated through a gateway **24**, which will effectively convert between circuit-switched and packet-switched voice information. Packet telephony may be facilitated via any number of devices, including a personal computer (PC) **25**, PDA **26**, and packet-based telephones, not shown. Notably, the personal computer **25**, as well as the PDA **26**, may be linked to the service node **30** via the Internet or like packet-switched network in traditional fashion. Further, access to the service node **30** may be provided by using a browser to interact with a web-based server implemented by the service node **30**. Alternatively, proprietary communication software and applications may be used to facilitate the interaction between these devices and the service node **30** to provide instructions for selecting call processing rules in a dynamic fashion.

Call processing, including call routing and control, is preferably provided via a signaling network **28**, such as the Signaling System 7 (SS7), which will directly or indirectly interact with the wireline switches **14**A and **14**B, wireless switch **14**C, and perhaps the gateway **24** to facilitate the establishment of telephony calls between various telephony devices. For implementation of the present invention, a call control entity referred to generally as a service node **30** will directly or indirectly through the signaling network **28** communicate with the wireline switches **14**A and **14**B, wireless switch **14**C, and the gateway **24**, to process incoming calls directed to any of the user's telephony devices that are illustrated in FIG. **1**. Alternatively, call processing signaling may be provided via a packet-based protocol such as SIP (Session Initiation Protocol, IETF standard RFC 3261) between the service node **30** and packet-based devices such as the gateway **24** or PC **25**. Depending on the capabilities of the various telephony switches **14**, the gateway **24**, and the signaling network **28**, optional signaling translators **32** may be provided to facilitate an interface between the service node **30** and the various telephony switches **14** and gateway **24**. The signaling translators **32** may provide protocol and signaling translations as necessary to enable the service node **30** and the various telephony switches **14** and gateway **24** to communicate. Further, an interactive voice response system (IVR) **36** may be provided to allow users to audibly, or via tones, configure or otherwise instruct the service node **30**.

In such a communication environment, the present invention provides a simple and effective way for a user to signal the service node **30** in a way that implements a desired call processing rule, and as such, have incoming calls initially intended for any one of the user's telephony devices directed to a defined telephony device, regardless of whether the telephony device is a wireline, wireless, or packet-based telephony device. In addition to simply vectoring calls to a desired telephony device, the call processing rules may also be more sophisticated, wherein calls may be directed to a common voicemail system **34**, or calls may be screened, blocked, or undergo like call handling functions. Preferably, signaling of the service node **30** is effected by user interaction with the telephony device or via the PC **25**, PDA **26**, or like interaction. For interactions with the telephony devices, the associated telephony switches **14** or gateways **24** will sense a particular user interaction or receive a signal from a user's telephony device, which will trigger sending a message to the service node **30** identifying the telephony device or user, along with the manner or type of interaction the user had with the telephony device. The interaction will define what call processing rules the service node **30** will apply for incoming calls directed to one of the user's associated telephony devices. Those skilled in the art will recognize numerous types of interactions and signals provided by the user via their telephony devices, and the following are merely exemplary to help illustrate the concepts of the present invention.

For wireline-based telephony device, such as the office telephone **16** and the home telephone **18**, the corresponding wireline switches **14** (A and B) may recognize an off hook followed by an on hook signal from the office telephone **16**

or home telephone **18** as a signal to send a message to the service node **30** indicative of that activity. The service node **30** will recognize that activity as the user selecting the particular device from which the off hook and on hook signals are received, and have all incoming calls directed to that particular telephony device. In a more sophisticated embodiment, the signal may simply select one of many call processing rules defined by the user. A confirmation of the implementation of the rule may be initiated by the service node **30** instructing the corresponding wireline switch **14** (A or B) to provide a special ring signal, illuminate a lamp on the telephony device, or provide a special dial tone the next time the telephone is taken off hook to indicate to the user that the call processing rule has been implemented.

The user may provide signals to the service node **30** in a wireless domain by simply entering a number or series of numbers followed by the send command. The wireless switch **14**C will send an appropriate message to the service node **30** upon receiving such an action, and the service node **30** will recognize the user based on the identification of the mobile terminal **20** and implement the desired call processing rule, which in its simplest form will have all incoming calls directed to any of the telephony devices of the user directed to the mobile terminal **20**. The service node **30** may effect confirmation by instructing the wireless switch **14**C to immediately provide a special ring signal, or initiate some type of message to the mobile terminal **20** indicative of implementation of the desired call processing rule. The messaging may also be an instant message, an email, a short message service (SMS) message, or the like. For either the wireline or wireless cases, the user may dial into the interactive voice response system **36**, which is capable of interacting with the service node **30** based on voice or touchtone input to establish as well as select call processing rules.

Alternatively, a computing device, such as the PC **25** or PDA **26**, may directly interact with the service node **30** through the Internet or like packet-switched network to establish or dynamically select a call processing rule. Preferably, an application running on the computing device will provide icons associated with the different call processing rules or devices to which incoming calls may be directed. Upon selecting or clicking on a desired icon, the application will send instructions to the service node **30** to implement the corresponding call processing rule. Confirmation of implementation of the call processing rule may be provided directly to the selected device or to the user via a pop-up message or like indication. Although it is preferred to confirm implementation of the desired call processing rule, it is not necessary. Further, confirmation may be provided to the other telephony devices as well as to the computing device, in addition to the particular device from which the instruction to select a call processing rules was initiated.

Importantly, the below illustrated example assumes that the office telephone **16**, home telephone **18**, personal computer **25**, and mobile terminal **20** are all devices having different directory numbers or network addresses and are all associated with a given user (or entity). The respective telephony switches **14**A, **14**B, and **14**C and gateway **24** are provisioned to effectively request call processing instructions from the service node **30**, and will process the call based on the response received from the service node **30**. The service node **30** will decide how to route the incoming call based on predefined call processing rules or rules previously established by the user, and the latest instructions received by the user for selecting a given one of the rules. The service node **30** may reroute the incoming call to another of the user's telephony devices, block the call, or forward the call to a voicemail system **34**, which will preferably be the only voicemail system **34** for the user, regardless of the telephony device for which the incoming call was originally intended. The call processing rules selected by the user may also be a function of virtually any criteria, such as time or date, line status, mobile terminal location, computer presence, an electronic calendar, the caller, or the called number, such that calls from different people may be handled in different ways based on any combination of those criteria. Further, the call processing may be based in part on the originally intended directory number dialed by the caller.

With reference to FIG. **2**, the basic functionality of a service node **30** is illustrated. Initially, the service node **30** will receive information, preferably from the user, to create the call processing rules (step **100**). The call processing rules will essentially be a profile identifying how to process an incoming call based on the chosen variables. From this set of call processing rules, the user can dynamically instruct the service node **30** to select a particular call processing rule via a telephony device, such as the office telephone **16**, home telephone **18**, or mobile terminal **20**, or alternately the personal computer **25** or PDA **26**. At this point, call processing is controlled by the default rule until instructions are received from the user via one of the telephony devices, PC **25**, or PDA **26** directing the service node **30** to apply a different call processing rule.

To initially configure the service node **30**, the user may access the service node **30** via any number of devices, including the PC **25**, PDA **26**, or telephony devices, directly or indirectly via an interactive voice response system **36** (see FIG. **1**). Those skilled in the art will recognize the multitude of ways for allowing a user to interact with the service node **30** to effectively establish the initial call processing rules and the actions to be received from the user, which will select the call processing rules. Interaction with the service node **30** via the computer **25** or PDA **26** is preferably done via a browser interface in traditional fashion. Preferably, the user is able to set up different types of profiles having different call processing rule sets, such that call processing is handled differently given the current state of the user. For example, different call processing rules may be established when the user is in the office, at home, telecommuting, traveling, on vacation, out of the office, or in a meeting. For additional information on establishing profiles and providing call processing rules, see U.S. application Ser. No. 10/382,247, entitled COMMON CALL ROUTING FOR MULTIPLE TELEPHONY DEVICES and filed Mar. 5, 2003, which is incorporated herein by reference in its entirety.

Once the information to create the call processing rules is received (step **100**), rules are then created based on the user information and a default rule is put into effect (step **102**). Preferably, the telephony switches **14** or gateways **24** are provisioned to interact with the service node **30** to receive call processing instructions upon receiving a call directed to a telephony device associated with the user. When the user wishes to implement a desired call processing rule, an appropriate action on the part of one of the associated telephony devices, PC **25**, or PDA **26** is initiated, which will directly or indirectly via a switching device send instructions including rule selection indicia to the service node **30**. The service node **30** will receive the selection indicia from the user and select a call processing rule based thereon (step **104**). Next, the service node **30** will effect delivery of confirmation indicia indicating the desired call processing

rule was selected and will be implemented until changed by the user or according to a previously defined call processing rule or profile (step **106**).

Once provisioned, the various telephony switches **14** and gateways **24** will recognize an incoming call intended for the user, and initiate interaction with the service node **30**. The service node **30** will receive the incoming call information from the telephony switches **14** or gateways **24** handling the incoming call (step **108**). The information received from the telephony switch **14** or gateway **24** is sufficient to identify the called party, and preferably the caller. As such, the service node **30** will initially identify the called party (step **110**) and then apply the call processing rules for the called party to determine how the telephony switch **14** or gateway **24** should handle the incoming call (step **112**). Next, the service node **30** will instruct the telephony switch **14** or gateway **24** to connect the call to the intended directory number, or to reroute the call to another directory number or to voicemail, based on the selected call processing rule (step **114**). The telephony switch **14** or gateway **24** will then process the incoming call accordingly. Notably, the use of directory number (DN) is intended to encompass any type of telephony addressing, including Internet Protocol (IP) addresses or the like, used for routing packet-based voice communications. If and as calls are forwarded to telephony switches **14** servicing the various telephony devices under the instruction of the service node **30**, each of the telephony switches **14** or gateways **24** will interact with the service node **30** as necessary to handle and direct the call according to the selected call processing rule.

An exemplary call flow is provided in FIG. **3**A, wherein an incoming call is originally intended for the user's office telephone **16**, wherein the user dynamically selects a call processing rule, which will at the service node **30** forward incoming calls directed to the office telephone **16**, home telephone **18**, or mobile terminal **20** to the mobile terminal **20**, which is associated with directory number DN **3**. As illustrated in FIG. **1**, the office telephone **16** is serviced by wireline switch A (**14**), the home telephone **18** is serviced by wireline switch B (**14**), and the mobile terminal **20** is serviced by wireless switch C (**14**).

Initially, the user will decide to have all incoming calls to the office telephone **16**, home telephone **18**, or mobile terminal **20** routed to the mobile terminal **20**, and as such will effectively initiate delivery of selection indicia from the mobile terminal **20** to the service node **30** by entering a defined number and hitting send. In this case, assume that entering the number 7 and hitting send will trigger the wireless switch **14**C to send a message to the service node **30**, triggering the service node **30** to implement a call processing rule to direct all incoming calls directed to any of the telephony devices to the mobile terminal **20**. Accordingly, when the user enters 7 and hits send, the selection indicia is sent from the mobile terminal **20** to wireless switch C (step **200**), which is provisioned to send an origination attempt trigger (O_A TRIG.) identifying the mobile terminal **20** and the selection indicia to the service node **30** (step **202**). The service node **30** will identify the user based on the identity of the mobile terminal **20**, preferably via the mobile terminal's caller ID or other identification indicia, access the call processing rules or profiles for the user, and select the appropriate call processing rule based on the selection indicia. Next, the service node **30** will send a confirmation message to wireless switch C (step **204**), which will trigger a special ring at the mobile terminal **20** or send a message to the mobile terminal **20** indicating that the instructions

have been implemented by the service node **30** and all calls will be forwarded to the mobile terminal **20** (step **206**).

For an incoming call to any of the telephony devices, and in this example, to the office telephone **16** using DN **1**, indication of the incoming call to the office telephone **16** is provided in an Initial Address Message (IAM) sent to wireline switch A from a telephony switch in the PSTN **12** (step **208**). Wireline switch A will recognize that the incoming call is intended for the office telephone **16** based on the office telephone's directory number DN **1** included in the IAM during incoming call processing (step **210**), and recognize that calls intended for the office telephone **16** should trigger the initiation of a termination attempt trigger message to the service node **30**. Accordingly, a termination attempt trigger message (T_A TRIG.) is sent to the service node **30** and will identify the directory number for the called party, DN **1**, and that of the caller, DN (step **212**). The service node **30** will directly or indirectly receive the termination attempt trigger, identify the selected call processing rule based on the notification of the called party (DN **1**), and determine how wireline switch A should process the incoming call (step **214**). In this case, the rule applied indicates wireline switch A should route the incoming call to directory number DN **3**, which is associated with the user's (called party's) mobile terminal **20**. As such, the service node **30** will send instructions to wireline switch A to forward the call to directory number DN **3** (step **216**). Wireline switch A will provide the requisite call processing (step **218**) and send an IAM directly or indirectly to wireless switch C, which services the mobile terminal **20** (step **220**).

Wireless switch C will provide the requisite call processing (step **222**), and recognize that incoming calls directed to the mobile terminal **20** require instructions from the service node **30**. As such, a termination attempt trigger is sent to the service node **30** identifying the caller by directory number DN, and called party by the directory number for the mobile terminal **20**, DN **3** (step **224**). The service node **30** will identify the call processing rule based on directory number DN **3**, recognize that the call should be routed to the mobile terminal **20** (step **226**) and send a continue message to wireless switch C (step **228**). Wireless switch C will cause the mobile terminal **20** to ring (step **230**), as well as send an Address Complete Message (ACM) to wireline switch A indicating that the home telephone **18** is ringing (step **232**). Wireline switch A will forward the ACM as necessary to the PSTN **12** and ultimately the originating switch (step **234**). When the mobile terminal **20** is answered, wireless switch C will recognize that the mobile terminal **20** has gone offhook (step **236**), which will trigger the sending of an Answer Message (ANM) to wireline switch A (step **238**), which will forward the ANM to the PSTN **12** in traditional fashion (step **240**). At this point, a voice connection is established between the mobile terminal **20** and the telephony device of the caller (step **242**).

Turning now to FIG. 3B, assume the incoming call from the caller is made directly to the mobile terminal **20** using directory number DN **3**. As such, the IAM is sent directly to wireless switch C (step **244**), which will provide the initial incoming call processing (step **246**). Upon recognizing that the call is intended for directory number DN **3** and requires support from the service node **30**, a termination attempt trigger is sent to the service node **30** identifying the directory numbers for the called party (user) and caller (step **248**). The service node **30** will identify the call processing rules to use based on the called party's directory number DN **3** and decide how wireless switch C should process the incoming call (step **250**). Since calls to the user are supposed to be

routed to the mobile terminal **20** associated with directory number DN **3** in this scenario, the service node **30** will cause wireless switch C to proceed in a normal fashion and route the call to the mobile terminal **20** by sending a continue message to wireless switch C (step **252**). Wireless switch C will proceed with call processing (step **254**) and begin ringing the mobile terminal **20** (step **256**). Concurrently, wireless switch C will send an ACM back to the PSTN **12** (step **258**) and await answering of the mobile terminal **20**. Upon being answered, wireless switch C will recognize that the mobile terminal **20** has gone offhook (step **260**) and send an ANM to the PSTN **12** (step **262**) in traditional fashion. At this point, a voice connection is established between the mobile terminal **20** and the telephony device of the caller (step **264**).

As seen from the above, allowing the telephony switches **14** or gateways **24** servicing various telephony devices of a given user to be coordinated for incoming calls to the user allows the user to effectively and efficiently control how calls are processed among any of the telephony devices, or the voicemail system **34**. The invention is particularly beneficial in allowing a user to effectively use her mobile terminal **20** in conjunction with her wireline work or home telephones **16**, **18**. By dynamically configuring the service node **30** to handle or otherwise process incoming calls to any of the telephony devices to a desired telephony device or a single voicemail system **34**, all incoming calls, regardless of the originally intended telephony device, are efficiently controlled by the user. If a common voicemail system **34** is defined, the user avoids having voicemails left in different voicemail systems or answering machines, while making communications with her much more efficient for those initiating the calls. By controlling call processing via the service node **30** for multiple telephony devices having unique directory numbers, the user does not have to use multiple call forwarding systems in a rudimentary manner to control call handling. Further, current call forwarding systems will not allow the associated telephony device to receive a call, and thus, call screening is not available.

A common scenario to most mobile terminal users is one where the user is in a meeting or participating in a conference call on an office telephone **16**. The service node **30** may have a profile for the user specifically adapted for meeting situations. Further, in these situations, most incoming calls should be diverted to voicemail, but there are certain people, such as the user's boss, that should be able to contact the user at any given time, even when in meetings. As such, the meeting profile in the service node **30** may be configured to divert incoming calls directed to the office telephone **16** or the mobile terminal **20** to a common voicemail system **34** from all calling parties, except the user's boss, whose various directory numbers are provided to the service node **30**. Thus, upon recognizing incoming calls to the office telephone **16** or mobile terminal **20** for the user, the service node **30** will direct the respective telephony switches **14** to forward the call to voicemail, unless the call is from the boss, in which case the call will be directed to the mobile terminal **20**.

The user may also have a general office profile that directs calls that are made to the mobile terminal **20**, office telephone **16**, or other associated telephony device to the office telephone **16**. To further refine this profile, the user may decide to block calls from certain callers and have those calls automatically forwarded to the voicemail system **34**. As those skilled in the art will recognize, various profiles based on user configuration, time and date, incoming or outgoing call identification, and the like, may be used to provide unlimited call processing rule sets, which may be dynamically selected by the user through her telephony devices, PC, PDA, or even via the IVR **36**. Each user may have any number of rule sets, and the service node **30** may support any number of telephony devices for a given user. Although a significant portion of the above disclosure is directed to wireline and wireless telephony communications, those skilled in the art will also recognize the ease with which call processing entities for packet-switched systems can interact with the service node **30** to facilitate the unified call processing of the present invention, and may support numerous users and their associated telephony devices. Notably, the term "user" used herein relates to any type of entity or individual, and the various telephony devices may be associated with multiple directory numbers.

The service node **30** may take on many forms and be integrated with other call processing systems, as well as having numerous interfaces for effectively communicating directly or indirectly with the various telephony switches **14**, gateways **24**, or other packet-based call processing entities. As illustrated in FIG. **4**, the service node **30** will typically be associated with a central processing unit (CPU) **38** having sufficient memory **40** for storing the necessary software **42** for implementing the concepts of the present invention. The CPU **38** will have a communication interface **44** for communicating directly or indirectly with the various telephony switches **14**, gateways **24**, or like call processing entities.

Those skilled in the art will recognize improvements and modifications to the preferred embodiments of the present invention. All such improvements and modifications are considered within the scope of the concepts disclosed herein and the claims that follow.

What is claimed is:

1. A method for allowing a user to control call processing comprising:

receiving selection indicia resulting from an action taken by the user at a user device;

implementing a call processing rule based on the selection indicia, the call processing rule defining how to process incoming calls directed to a plurality of telephony devices associated with the user, wherein each of the plurality of telephony devices is associated with a unique address or telephony number;

sending instructions to process an incoming call intended for one of the plurality of telephony devices, the incoming call bearing a destination address of the unique address or telephony number and the instructions to process the incoming call are based on the call processing rule; and

sending confirmation indicia intended to effect delivery of an alert to the user indicative of the call processing rule being implemented.

2. The method of claim **1** wherein the user device is a wireline telephony device, which is one of the plurality of telephony devices, and the selection indicia is received from a wireline telephony switch, which sends the selection indicia in response to an action taken by the wireline telephony device.

3. The method of claim **2** wherein sending confirmation indicia comprises sending the confirmation indicia to the wireline telephony switch, which is configured to provide the alert to the user via the wireline telephony device.

4. The method of claim **2** wherein the instructions are sent to the wireline telephony switch.

5. The method of claim **2** wherein the action taken by the wireline telephony device is toggling the wireline telephony device off and on book.

6. The method of claim 1 wherein the user device is a wireless telephony device, which is one of the plurality of telephony devices, and the selection indicia is received from a wireless telephony switch, which sends the selection indicia in response to an action taken by the wireless telephony device.

7. The method of claim 6 wherein sending confirmation indicia comprises sending the confirmation indicia to the wireless telephony switch, which is configured to provide the alert to the user via the wireless telephony device.

8. The method of claim 6 wherein the instructions are sent to the wireless telephony switch.

9. The method of claim 6 wherein the action taken by the wireless telephony device is entering and sending a defined number.

10. The method of claim 1 wherein the user device is a packet-based telephony device, which is one of the plurality of telephony devices, and the selection indicia is received from a packet-based gateway, which sends the selection indicia in response to an action taken by the packet-based telephony device.

11. The method of claim 10 wherein sending confirmation indicia comprises sending the confirmation indicia to the packet-based gateway, which is configured to provide the alert to the user via the packet-based telephony device.

12. The method of claim 10 wherein the instructions are sent to the packet-based gateway.

13. The method of claim 1 wherein the user device is a personal computing device and the selection indicia is received from the personal computing device.

14. The method of claim 1 further comprising receiving the selection indicia resulting from actions taken by the user at a plurality of user devices serviced by different telephony switches and sending the instructions to process the incoming calls intended for the plurality of telephony devices based on the call processing rule.

15. The method of claim 11 wherein one of the different telephony switches is a wireless telephony switch and another one of the different telephony switches is a wireline telephony switch.

16. The method of claim 1 wherein the call processing rule effects routing of all incoming calls intended for any of the plurality of telephony devices to the user device, which is one of the plurality of telephony devices associated with the user.

17. The method of claim 1 wherein the call processing rule effects routing of all incoming calls intended for any of the plurality of telephony devices to another user device, which is one of the plurality of telephony devices associated with the user.

18. The method of claim 1 wherein the call processing rule effects routing of all incoming calls intended for any of the plurality of telephony devices, which is one of the plurality of telephony devices associated with the user, to a voicemail system associated with the user.

19. The method of claim 1 wherein the alert is a message sent to the user device.

20. The method of claim 1 wherein the alert is a special dial tone provided when the user uses the user device.

21. The method of claim 1 wherein the alert is lighting a light on the user device.

22. The method of claim 1 wherein the alert is a message provided to the user via a personal computing device of the user.

23. A system for allowing a user to control call processing comprising:

a) an interface; and

b) a central processing unit associated with the interface and adapted to:

receive selection indicia resulting from an action taken by the user at a user device;

implement a call processing rule based on the selection indicia, the call processing rule defining how to process incoming calls directed to a plurality of telephony devices associated with the user, wherein each of the plurality of telephony devices is associated with a unique address or telephony number;

send instructions to process an incoming call intended for one of the plurality of telephony devices, the incoming call bearing a destination address of the unique address or telephony number and the instructions to process the incoming call are based on the call processing rule; and

send confirmation indicia intended to effect delivery of an alert to the user indicative of the call processing rule being implemented.

24. The system of claim 23 wherein the user device is a wireline telephony device, which is one of the plurality of telephony devices, and the selection indicia is received from a wireline telephony switch, which sends the selection indicia in response to an action taken by the wireline telephony device.

25. The system of claim 24 wherein the central processing unit is further adapted to send the confirmation indicia to the wireline telephony switch, which is configured to provide the alert to the user via the wireline telephony device.

26. The system of claim 24 wherein the instructions are sent to the wireline telephony switch.

27. The system of claim 24 wherein the action taken by the wireline telephony device is toggling the wireline telephony device off and on hook.

28. The system of claim 23 wherein the user device is a wireless telephony device, which is one of the plurality of telephony devices, and the selection indicia is received from a wireless telephony switch, which sends the selection indicia in response to an action taken by the wireless telephony device.

29. The system of claim 28 wherein the central processing unit is further adapted to send the confirmation indicia to the wireless telephony switch, which is configured to provide the alert to the user via the wireless telephony device.

30. The system of claim 28 wherein the instructions are sent to the wireless telephony switch.

31. The system of claim 28 wherein the action taken by the wireless telephony device is entering and sending a defined number.

32. The system of claim 23 wherein the user device is a packet-based telephony device, which is one of the plurality of telephony devices, and the selection indicia is received from a packet-based gateway, which sends the selection indicia in response to an action taken by the packet-based telephony device.

33. The system of claim 32 wherein the central processing unit is further adapted to send the confirmation indicia to the packet-based gateway, which is configured to provide the alert to the user via the packet-based telephony device.

34. The system of claim 32 wherein the instructions are sent to the packet-based gateway.

35. The system of claim 23 wherein the user device is a personal computing device and the selection indicia is received from the personal computing device.

36. The system of claim 23 wherein the central processing unit is further adapted to receive selection indicia resulting from actions taken by the user at a plurality of user devices

serviced by different telephony switches and sending instructions to process incoming calls intended for the plurality of telephony devices based on the call processing rule.

37. The system of claim 36 wherein one of the different telephony switches is a wireless telephony switch and another one of the different telephony switches is a wireline telephony switch.

38. The system of claim 23 wherein the call processing rule effects routing of all incoming calls intended for any of the plurality of telephony devices to the user device, which is one of The plurality of telephony devices associated with the user.

39. The system of claim 23 wherein the call processing rule effects routing of all incoming calls intended for any of the plurality of telephony devices to another user device, which is one of the plurality of telephony devices associated with the user.

40. The system of claim 23 wherein the call processing rule effects routing of all incoming calls intended for any of the plurality of telephony devices, which is one of the plurality of telephony devices associated with the user, to a voicemail system associated with the user.

41. The system of claim 23 wherein the alert is a message sent to the user device.

42. The system of claim 23 wherein the alert is a special dial tone provided when the user uses the user device.

43. The system of claim 23 wherein the alert is lighting a light on the user device.

44. The system of claim 23 wherein the alert is a message provided to the user via a personal computing device of the user.

* * * * *

EXHIBIT H

US006934279B1

(12) **United States Patent**
Sollee et al.

(10) **Patent No.:** **US 6,934,279 B1**
(45) **Date of Patent:** **Aug. 23, 2005**

(54) **CONTROLLING VOICE COMMUNICATIONS OVER A DATA NETWORK**

(75) Inventors: **Patrick N. Sollee**, Richardson, TX (US); **David R. Creech**, Carrollton, TX (US); **Gregory T. Osterhout**, Coppell, TX (US); **Christopher L. Jessen**, McKinney, TX (US)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/524,342**

(22) Filed: **Mar. 13, 2000**

(51) **Int. Cl.**^{7} **H04L 12/56**
(52) **U.S. Cl.** **370/352**; 370/356
(58) **Field of Search** 370/352, 465, 370/467, 356, 260; 379/230, 15.01; 709/223

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,706,242 A		11/1987	Harland	370/60
6,084,956 A	*	7/2000	Turner et al.	370/467
6,091,808 A		7/2000	Wood et al.	379/201
6,185,565 B1		2/2001	Meubus et al.	707/10
6,192,050 B1		2/2001	Stovall	370/389
6,337,858 B1	*	1/2002	Petty et al.	370/356
6,363,065 B1	*	3/2002	Thornton et al.	370/352
6,404,774 B1	*	6/2002	Jenness	370/465
6,421,424 B1	*	7/2002	Creamer et al.	379/15.01
6,430,282 B1		8/2002	Bannister et al.	379/211.02
6,526,129 B1		2/2003	Beaton et al.	379/90.01
6,549,621 B1	*	4/2003	Christie et al.	379/230
6,570,870 B1	*	5/2003	Berstis	370/356
6,614,899 B1		9/2003	Sollee et al.	379/218.01
6,671,262 B1	*	12/2003	Kung et al.	370/260
6,687,747 B1	*	2/2004	Huang et al.	709/223

FOREIGN PATENT DOCUMENTS

EP	0 721 266 A2		7/1996	
EP	0 829 995 A2		3/1998	
EP	0 836 295 A2		4/1998	
EP	1109368 A2	*	6/2001	H04L/12/64
WO	WO 99/05590		2/1999	

OTHER PUBLICATIONS

Steve Donovan, et al., *The SIP Info Method,* Internet Draft, pp. 1–7 (Jun. 1999).

Steve Donovan, *The SIP Info Method*, Internet Draft, pp. 1–10 (Feb. 2000).

Jon Crowcroft, *Comments About H.323 and SIP,* pp. 1–6, printed from web site http://www.cs.columbia.edu (Jan. 22, 1998).

Comparison of H.323 and SIP, pp. 1–5, printed from web site http://www.cs.columbia.edu, dated at least as early as Feb. 3, 2000.

Information Sciences Institute, *Internet Protocol, Darpa Internet Program Protocol Specification, RFC 791*, pp. 1–48 (Sep. 1981).

J. Postel, *User Datagram Protocol, RFC 768,* pp. 1–3 (Aug. 1980).

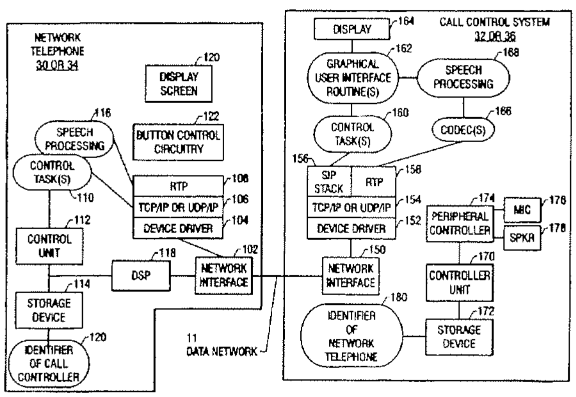
(Continued)

Primary Examiner—Salvatore Cangialosi
(74) *Attorney, Agent, or Firm*—Trop, Pruner & Hu, P.C.

(57) **ABSTRACT**

A method and apparatus of communicating over a data network includes providing a user interface in a control system for call control and to display information relating to a call session. The control system communicates one or more control messages (e.g., Session Initiation Protocol or SIP messages) over the data network to establish a call session with a remote device in response to receipt of a request through the user interface. One or more commands are transmitted to a voice device associated with the control system to establish the call session between the voice device and the remote device over the data network. A Real-Time Protocol (RTP) link may be established between the voice device and the remote device.

39 Claims, 7 Drawing Sheets



OTHER PUBLICATIONS

M. Handley, et al., *SDP: Session Description Protocol, RFC 2327,* pp. 1–39 (Apr. 1998).
H. Schulzrinne, et al., *RTP: A Transport Protocol for Real–Time Applications, RFC 1889,* pp. 1–69 (Jan. 1996).
S. Deering, et al., *Internet Protocol, Version 6 (IPv6) Specification, RFC 2460,* pp. 1–36 (Dec. 1998).
M. Handley, et al., *SIP: Session Initiation Protocol, RFC 2543,* pp. 1–153 (Mar. 1999).
Adam Roach, *Event Notification in SIP,* Internet Draft, pp. 1–8 (Mar. 2000).
Scott Petrack, et al., *The Pint Service Protocol: Extensions to SIP and SDP for IP Access to Telephone Call Services,* Internet Draft, pp. 1–59 (Oct. 14, 1999).
Jiri Kuthan, *Sample Uses of SIP Info with Varying Reliability Needs,* Internet Draft, pp. 1–7 (Oct. 17, 1999).
Robert Sparks, et al., *SIP Telephony Service Examples with Call Flows,* Internet Draft, pp. 1–79 (Oct. 1999).
U.S. Appl. No. 09/307,356, *Telephony and Data Network Services at a Telephone,* pp. 1–113, Figures 1–6, filed May 7, 1999.
G.A. Thom, *H. 323: The Multimedia Communications Standard For Local Area Networks,* IEEE Communications Magazine, IEEE Service Center, N.J., vol. 34, No. 12, Dec. 1996, pp. 52–56.

* cited by examiner

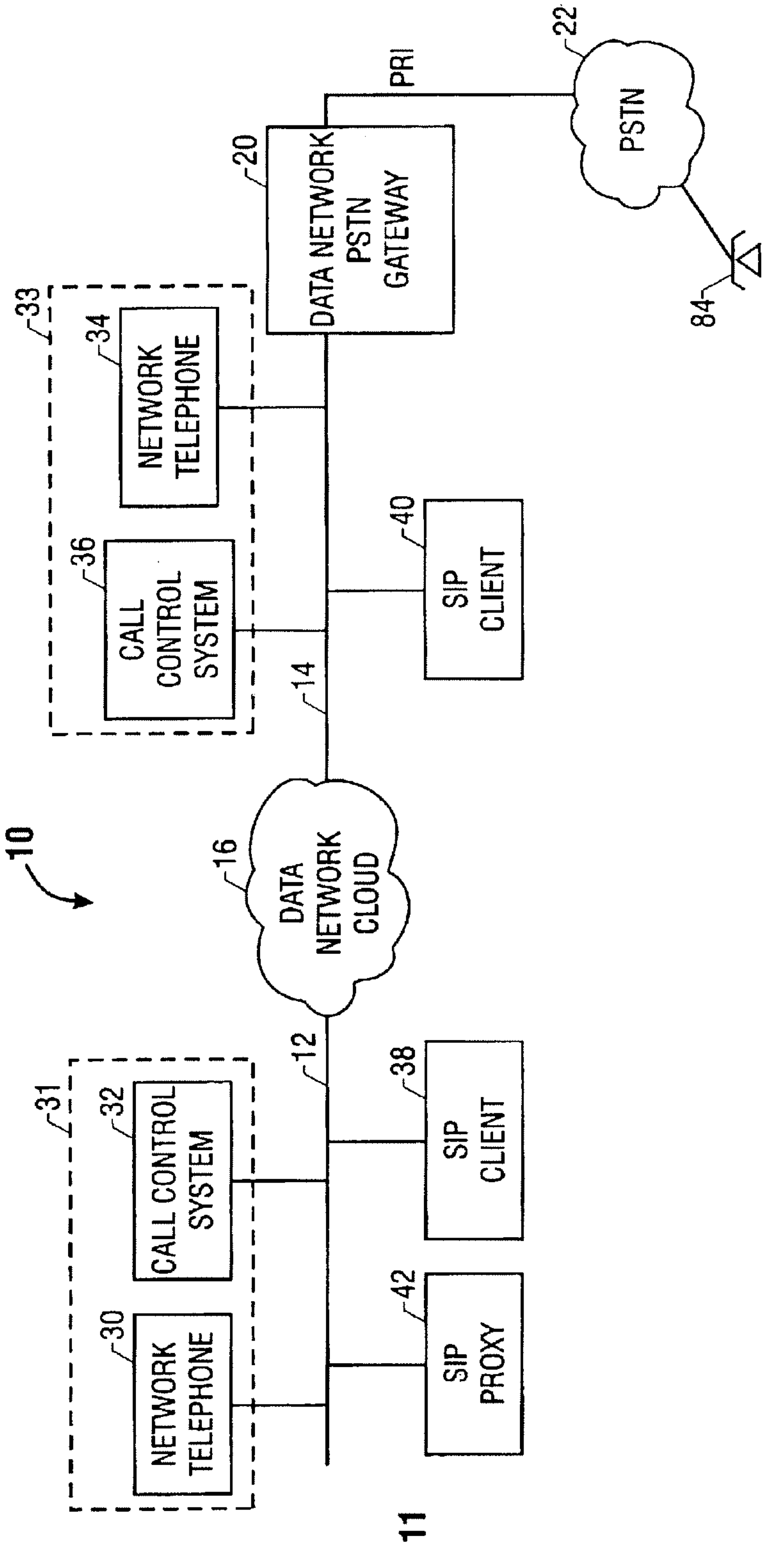

10
11
NETWORK TELEPHONE
30
CALL CONTROL SYSTEM
32
31
SIP PROXY
42
SIP CLIENT
38
12
DATA NETWORK CLOUD
16
14
CALL CONTROL SYSTEM
36
NETWORK TELEPHONE
34
33
SIP CLIENT
40
DATA NETWORK PSTN GATEWAY
20
PRI
PSTN
22
84

FIG. 1

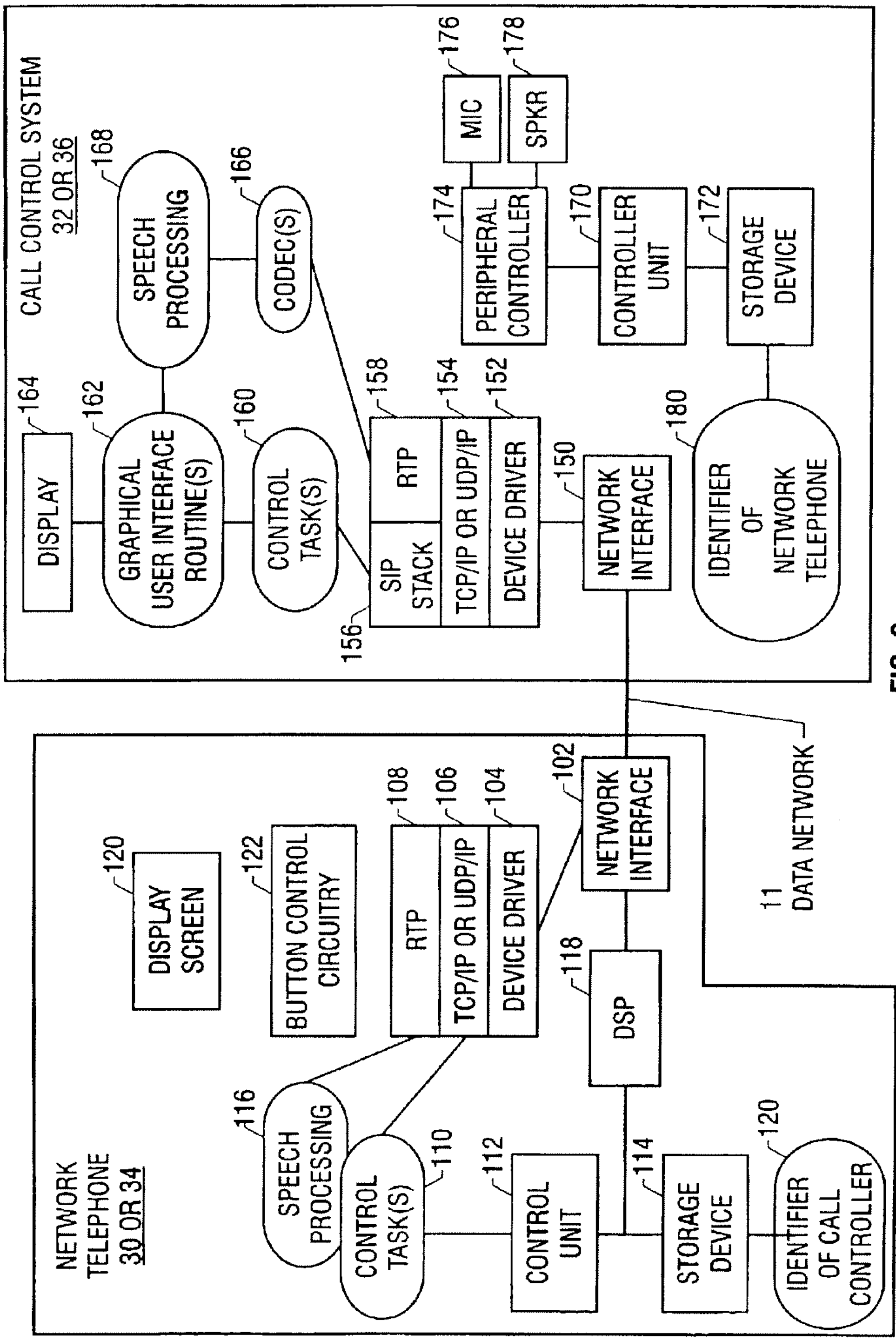

CALL CONTROL SYSTEM
32 OR 36
DISPLAY
164
GRAPHICAL USER INTERFACE ROUTINE(S)
162
SPEECH PROCESSING
168
CODEC(S)
166
CONTROL TASK(S)
160
SIP STACK
156
RTP
158
TCP/IP OR UDP/IP
154
DEVICE DRIVER
152
NETWORK INTERFACE
150
IDENTIFIER OF NETWORK TELEPHONE
180
MIC
176
SPKR
178
PERIPHERAL CONTROLLER
174
CONTROLLER UNIT
170
STORAGE DEVICE
172
DATA NETWORK
11
NETWORK TELEPHONE
30 OR 34
DISPLAY SCREEN
120
BUTTON CONTROL CIRCUITRY
122
RTP
108
TCP/IP OR UDP/IP
106
DEVICE DRIVER
104
NETWORK INTERFACE
102
DSP
118
SPEECH PROCESSING
116
CONTROL TASK(S)
110
CONTROL UNIT
112
STORAGE DEVICE
114
IDENTIFIER OF CALL CONTROLLER
120

FIG. 2

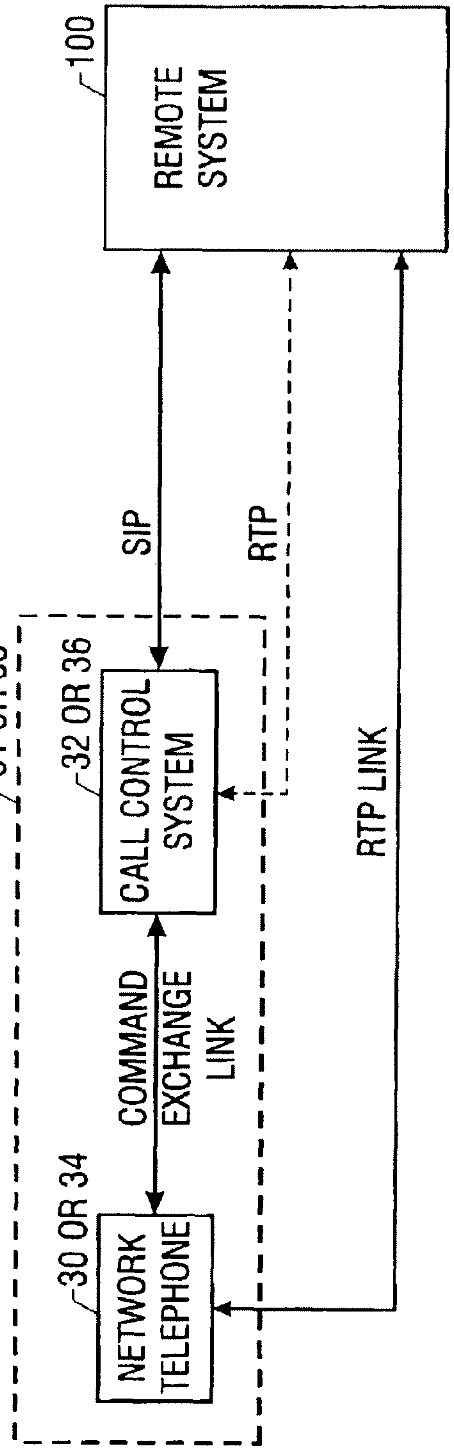

100
REMOTE SYSTEM
SIP
RTP
31 OR 33
32 OR 36
CALL CONTROL SYSTEM
RTP LINK
COMMAND EXCHANGE LINK
30 OR 34
NETWORK TELEPHONE

FIG. 3

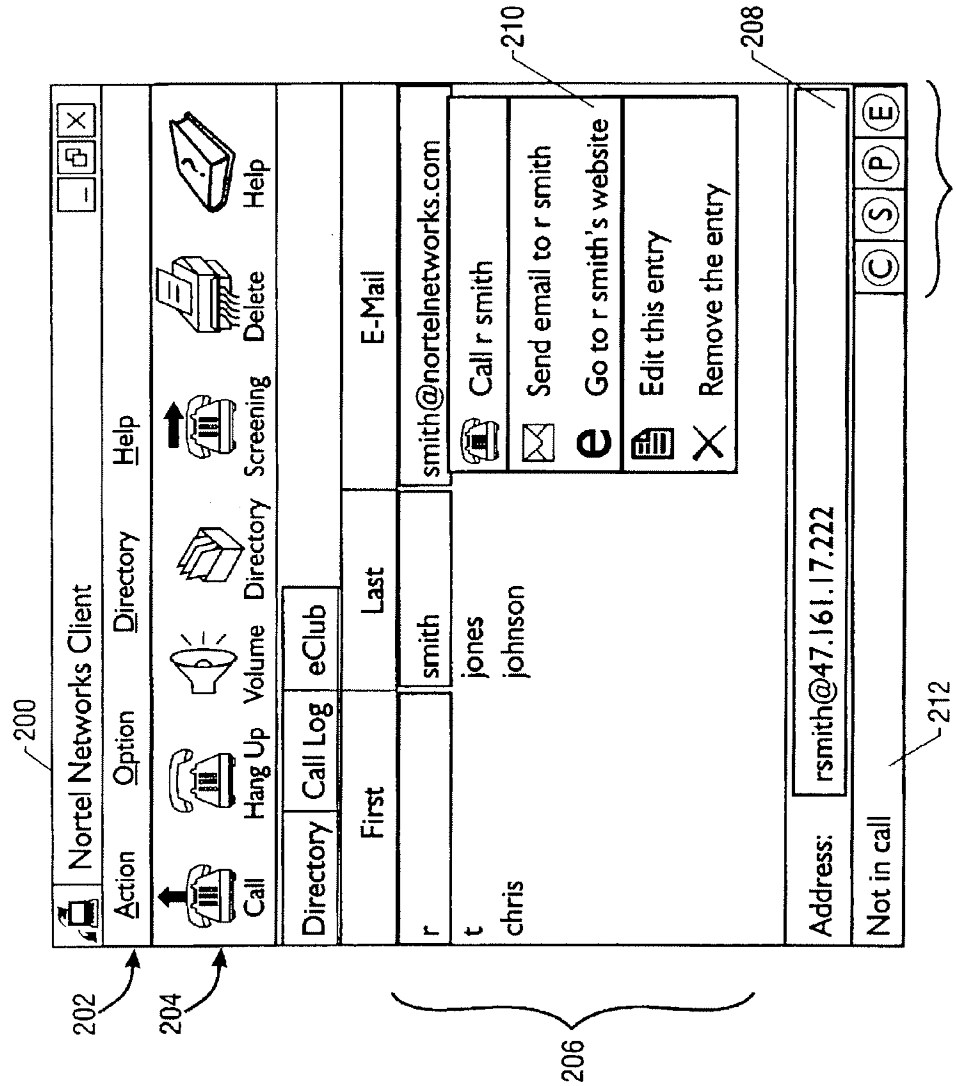

200
Nortel Networks Client
Action
Option
Directory
Help
Call
Hang Up
Volume
Directory
Screening
Delete
Help
Directory
Call Log
eClub
First
Last
E-Mail
r
smith
smith@nortelnetworks.com
t
jones
chris
johnson
Call r smith
Send email to r smith
Go to r smith's website
Edit this entry
Remove the entry
210
208
Address:
rsmith@47.161.17.222
Not in call
C
S
P
E
202
204
206
212
214

FIG. 4

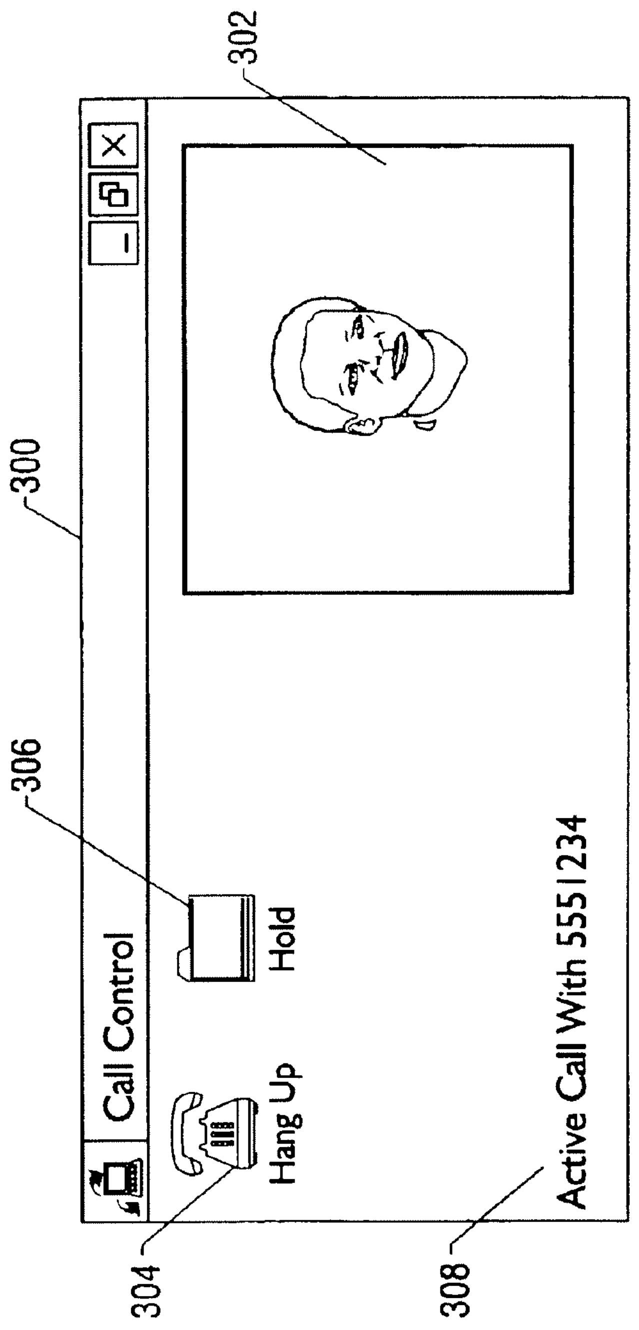

300
302
306
Call Control
Hang Up
Hold
Active Call With 5551234
304
308

FIG. 5

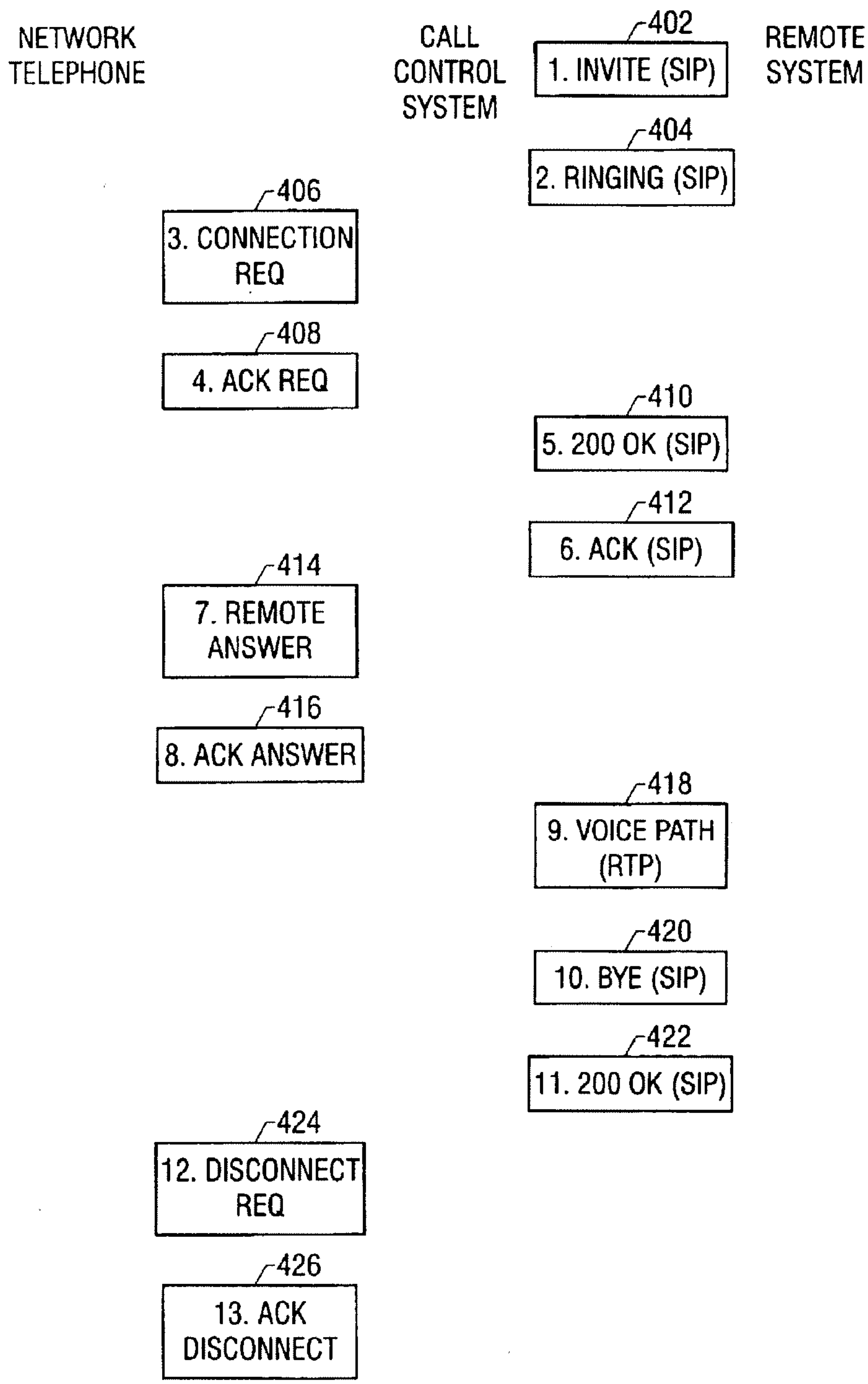

NETWORK TELEPHONE
CALL CONTROL SYSTEM
REMOTE SYSTEM
402
1. INVITE (SIP)
404
2. RINGING (SIP)
406
3. CONNECTION REQ
408
4. ACK REQ
410
5. 200 OK (SIP)
412
6. ACK (SIP)
414
7. REMOTE ANSWER
416
8. ACK ANSWER
418
9. VOICE PATH (RTP)
420
10. BYE (SIP)
422
11. 200 OK (SIP)
424
12. DISCONNECT REQ
426
13. ACK DISCONNECT

FIG. 6

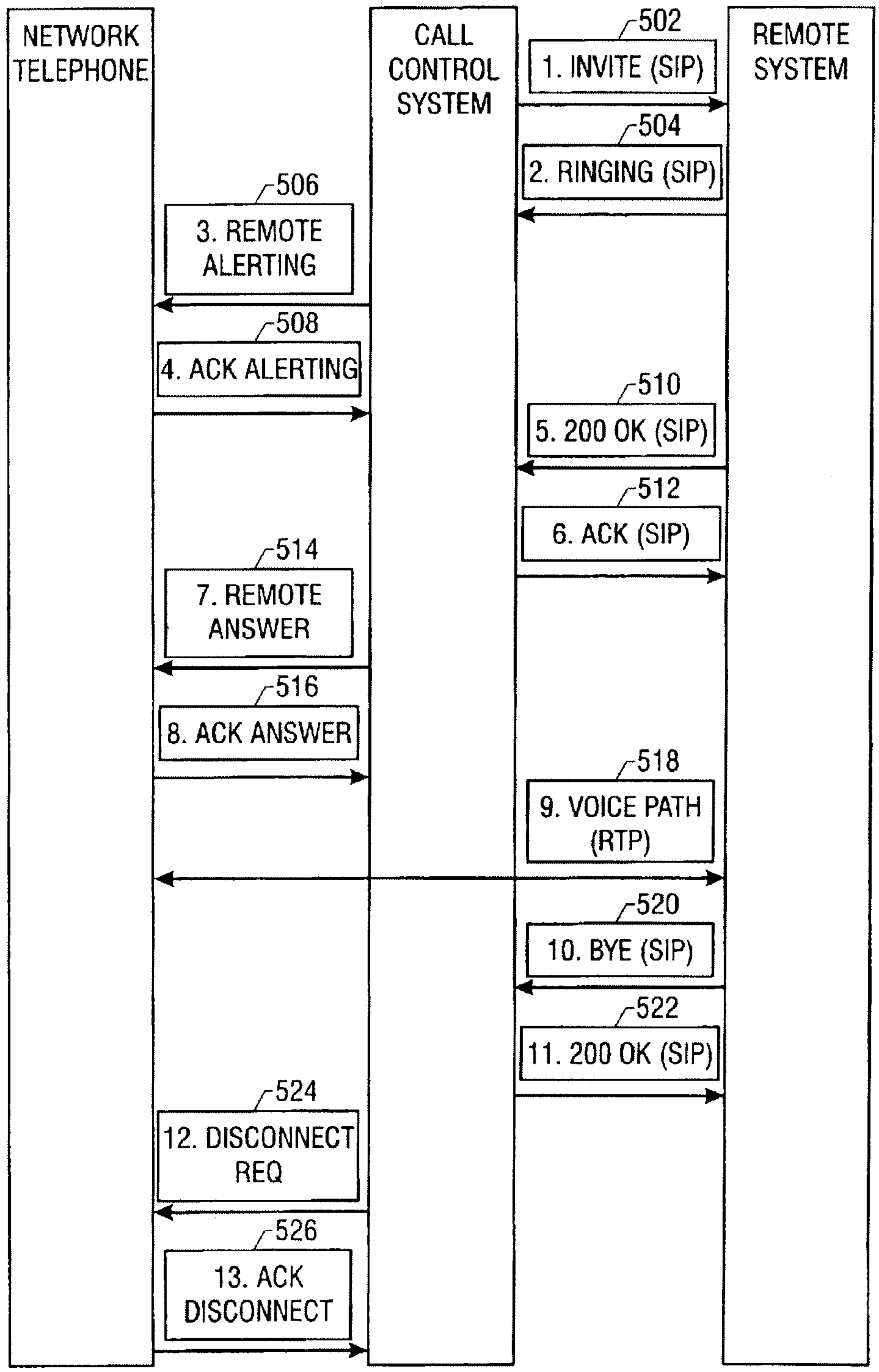

NETWORK TELEPHONE
CALL CONTROL SYSTEM
REMOTE SYSTEM
502
1. INVITE (SIP)
504
2. RINGING (SIP)
506
3. REMOTE ALERTING
508
4. ACK ALERTING
510
5. 200 OK (SIP)
512
6. ACK (SIP)
514
7. REMOTE ANSWER
516
8. ACK ANSWER
518
9. VOICE PATH (RTP)
520
10. BYE (SIP)
522
11. 200 OK (SIP)
524
12. DISCONNECT REQ
526
13. ACK DISCONNECT

FIG. 7

CONTROLLING VOICE COMMUNICATIONS OVER A DATA NETWORK

BACKGROUND

The invention relates to controlling voice communications over a data network.

Data networks are widely used to link various types of network elements, such as personal computers, servers, gateways, network telephones, and so forth. Data networks may include private networks (such a local area networks or wide area networks) and public networks (such as the Internet). Popular forms of communications between network elements across such data networks include electronic mail, file transfer, web browsing, and other exchanges of digital data.

With the increased capacity and reliability of data networks, voice communications (including telephone calls, video conferencing, and so forth) over data networks have become possible. Voice communications over data networks are unlike voice communications in a conventional public switched telephone network (PSTN), which provides users with dedicated, end-to-end circuit connections for the duration of each call. Communications over data networks, such as IP (Internet Protocol) networks, are performed using packets or datagrams that are sent in bursts from a source to one or more destination nodes. Voice data sent over a data network typically shares network bandwidth with conventional non-voice data (e.g., data associated with electronic mail, file transfer, web access, and other traffic).

Various standards have been proposed for voice and multimedia communications over data networks. One such standard is the H.323 Recommendation from the International Telecommunications Union (ITU), which describes terminals, equipment, and services for multimedia communications over data networks.

Another standard for voice and multimedia communications is the Session Initiation Protocol (SIP), which establishes, maintains, and terminates multimedia sessions over a data network. SIP is part of a multimedia data and control architecture developed by the Internet Engineering Task Force (IETF). The IETF multimedia data and control architecture also includes the Resource Reservation Protocol (RSVP) for reserving network resources; the Real-Time Transport Protocol (RTP) for transporting real-time data and providing quality of service (QoS) feedback; the Real-Time Streaming Protocol (RTSP) for controlling delivery of streaming media; the Session Announcement Protocol (SAP) for advertising multimedia sessions by multicast; and the Session Description Protocol (SDP) for describing multimedia sessions.

To perform voice communications over a data network, a typical computer system (such as a desktop computer system or a portable computer system) may be equipped with voice processing capabilities. Such capabilities include a microphone, ear phones or speakers, and speech processing software. Typically, the speech processing software includes coder/decoders (CODECs) to encode and decode voice data. The voice processing software, including the CODECs, may be run on a microprocessor of a typical computer system. However, due to the intensive data processing typically required to process voice data, speech performance may not be optimum. For example, there may be delays associated with the transfer of such voice data due to the amount of time needed to process the voice data. Also, if certain types of CODECs that have less resource requirements are selected, voice quality may suffer.

Also, the computer system needs to be fitted with speakers, microphones, and sound cards to enable speech processing. Further, such speakers, microphones, and sound cards may not provide the desired level of quality, or if they do, may be relatively expensive. Additionally, to add such speech processing components to a computer system may require some configuration to be performed by a user, a process that an unsophisticated user may have difficulty with.

Unless a computer system with powerful processing capabilities are provided, the voice quality provided by such computer systems are not at the level typically experienced (and expected) by users of standard telephones. Such "standard" telephones may include analog telephones coupled to a local or central switching office or digital telephones coupled to a private branch exchange (PBX) system. More recently, network telephones have been developed that are capable of being connected directly to a data network, such as an IP network. These network telephones are capable of placing telephony calls over a data network. The voice quality offered by such telephones are typically superior to those that can be offered by computer systems, since such network telephones typically include dedicated digital signal processors (DSPs) that perform the data intensive calculations involved in speech processing. However, the existing network telephones do not provide desired multimedia presentation capabilities such as those offered by displays of computer systems. Thus, while network telephones offer superior speech capabilities, it does have the desired multimedia capabilities. On the other hand, computer systems have superior multimedia capabilities, but they suffer from relatively poor speech processing performance.

A need thus exists for an improved method and apparatus for controlling voice communications over data networks.

SUMMARY

In general, according to one embodiment, a method of communicating over a data network includes communicating, in a control system, one or more control messages over the data network to establish a call session with a remote device coupled to the data network. One or more commands are transmitted to a voice device coupled to the data network. The call session between the voice device and the remote device is established over the data network. Information associated with the call session is displayed on the control system.

In general, according to another embodiment, a method of communicating over a data network includes providing a user interface in a control system for establishing call sessions. One or more control messages are communicated by the control system over the data network to establish a call session with a remote device in response to receipt of a request through the user interface. One or more commands are transmitted to a voice device associated with a control system to establish the call session between the voice device and the remote device over the data network.

Some embodiments of the invention may include one or more of the following advantages. The voice processing capabilities of a voice device, such as a network telephone, may be advantageously used to provide superior voice quality, while at the same time, a control system such as a computer may be used to provide a convenient user interface for the user to perform call control and to view status and other information relating to the call session. Thus, voice quality associated with call sessions over data networks such as packet-switched data networks is enhanced using embodiments of the invention.

Other features and advantages will become apparent from the following description, from the drawings, and from the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is block diagram of an embodiment of a communications system.

FIG. **2** illustrates components in a network telephone and a call control system in accordance with an embodiment.

FIG. **3** illustrates control and data paths between network elements used during a call session in accordance with one embodiment.

FIGS. **4** and **5** illustrate example screens displayed by the call control system of FIG. **2** in accordance with an embodiment.

FIG. **6** is a message flow diagram of messages exchanged between network elements in the communications system of FIG. **1** for processing an incoming call.

FIG. **7** is a message flow diagram of messages exchanged between network elements in the communications system of FIG. **1** for placing an outgoing call.

DETAILED DESCRIPTION

In the following description, numerous details are set forth to provide an understanding of the present invention. However, it will be understood by those skilled in the art that the present invention may be practiced without these details and that numerous variations or modifications from the described embodiments may be possible. For example, although reference is made to Session Initiation Protocol (SIP) communications sessions in accordance with some embodiments, other protocols may be performed in further embodiments.

Referring to FIG. **1**, a communications system **10** includes a first data network **12** and a second data network **14** that are coupled through a data network cloud **16**. The data network cloud **16** may include various links, communications paths, and routers for routing messages between data networks **12** and **14**. The data network cloud **16** may include a public network such as the Internet. The data networks **12** and **14** may be private networks such as local area networks (LANs) or wide area networks (WANs). In the ensuing discussion, one or some combination of the data networks **12** and **14** and data network cloud **16** may be referred to collectively as the data network **11**. As used here, a "data network" or "network" may refer to one or more communications networks, channels, links, or paths and systems (such as routers) used to route data over such networks, channels, links, or paths.

The data network **11** may include an Internet Protocol (IP) network, which is a packet-switched network. One version of IP is described in Request for Comments (RFC) 791, entitled "Internet Protocol," dated September 1981. Other versions of IP, such as IPv6, or other connectionless, packet-switched standards may also be utilized in further embodiments. A version of IPv6 is described in RFC 2460, entitled "Internet Protocol, Version 6 (IPv6) Specification," dated December 1998. Packet-switched data networks such as IP networks communicate with packets, datagrams or other units of data over the data networks. Unlike circuit-switched networks, which provide a dedicated end-to-end connection or physical path for the duration of a call session, a packet-switched network is one in which the same path may be shared by several network elements. Packet-switched networks such as IP networks are based on a connectionless internetwork layer. Packets or other units of data injected into a packet-switched data network may travel independently over any path (and possibly over different paths) to a destination point. The packets may even arrive out of order. Routing of the packets is based on one or more addresses carried in each packet.

The packet-based network **12** may also be connection-oriented, such as an ATM (Asynchronous Transfer Mode) network or a Frame Relay network. In a connection-oriented, packet-based network, a virtual circuit or connection is established between two end points. In such connection-oriented networks, packets are received in the same order in which they were transmitted.

Network elements connected to the data network **11** may also be coupled through a data network-PSTN gateway **20** to a public-switched telephone network (PSTN) **22**. The link between the gateway **20** and the PSTN **22** may be a primary rate interface (PRI) link according to ISDN (Integrated Services Digital Network). Standard non-data network telephones **24** may be coupled to the PSTN **22**. Call sessions can thus be established between a data network element and one of telephones **84**.

In the example embodiment as illustrated in FIG. **1**, audio (e.g., voice) and multimedia (e.g., audio and video) communications may occur over the data network **11** between or among various network elements, including network telephones **30** and **34** and call control systems **32** and **36**. Other devices capable of voice or multimedia sessions include SIP (Session Initiation Protocol) client systems **38** and **40**. The SIP client systems **38** and **40** are capable of communicating using SIP messaging to establish call sessions. As used here, a "call session" refers generally to either a voice or a multimedia session established between two or more elements coupled to the data network **11** (or any other packet-switched data network). SIP is part of the multimedia data and control architecture from the Internet Engineering Task Force (IETF). A version of SIP is described in RFC 2543, entitled "SIP: Session Initiation Protocol," dated August 1999. SIP may be used to initiate call sessions as well as to invite members to a session that may have been advertised by some other mechanism, such as electronic mail, news groups, web pages, and other mechanisms. The other protocols in the IETF multimedia and control architecture include the Resource Reservation Protocol (RSVP), as described in RFC 2205; the Real-Time Transport Protocol (RTP), as described in RFC 1889; the Real-Time Streaming Protocol (RTSP), as described in RFC 2326; the Session Description Protocol (SDP), as described in RFC 2327; and the Session Announcement Protocol (SAP).

Other standards may be employed in further embodiments for controlling call sessions over the data network **11**. Such other standards may be any other standard that provides for interactive, real-time voice communications over the data network.

The SIP client systems **38** and **40** as shown in FIG. **1** include client application programs that are capable of sending SIP requests to perform call requests. The systems **38** and **40** may also be SIP servers. A server according to SIP may be an application program that accepts SIP requests to service calls and to send back responses to SIP requests. Thus, a system can be either a SIP client or a SIP server. A SIP proxy system, such as system **42**, may include an intermediary program that acts as both a server and a client for making requests on behalf of other clients.

In the system **10** as shown in FIG. **1**, the call control systems **32** and **36** are SIP-enabled; that is, the call control systems **32** and **36** are capable of sending and accepting SIP

requests to establish call sessions. The call control systems 32 and 36 may be implemented on a standard computer system platform. Unlike the call control systems 32 and 36, however, the network telephones 30 and 34 are not SIP-enabled in one embodiment. Although they are capable of communicating audio data over the data network 11, the network telephones 30 and 34 are not enabled to send or accept SIP messages (or other types of messages for establishing interactive, real-time voice communications) to establish call sessions. In accordance with some embodiments, the establishment, management, and termination of call sessions are controlled by the call control systems 32 and 36. Thus, the call control system 32 makes SIP requests on behalf of the network telephone 30, while the call control system 36 makes SIP requests on behalf of the network telephone 34. Once a call session is established, the network telephone 30 or 34 participates in the communication of voice data over the network 11.

By employing the arrangement as shown in FIG. 1, the superior voice capabilities of network telephones 30 and 34 may be utilized to provide enhanced voice quality for users making telephony calls over the data network 11. At the same time, associated call control systems 32 and 36 are used to provide call signaling communications and to provide the user with a convenient user interface to perform call control as well as display information associated with the call session.

The call control system 32 and the network telephone 30 may be collectively referred to as a telephony system 31. Similarly, the call control system 36 and network telephone 34 may be collectively referred to as a telephony system 33. To establish a call session between the telephony system 31 or 33 and another SIP-enabled remote system 100, as shown in FIG. 3, the call control system 32 or 36 sends SIP messages to the remote system 100 to establish a call session. The remote system 100 may be any system or device on the data network 11 that is capable of participating in a SIP-established call session. The call control system 32 or 36 also exchanges commands according to a predetermined format with the network telephone 30 or 34 to let the network telephone 30 or 34 know of the current status of the call setup. Once a call is established, a link may be established between the network telephone 30 or 34 and the remote system 100 over the data network 11. The link may be a Real-Time Protocol (RTP) link to communicate with voice data. Thus, in the telephony system 31 or 33, the call control system 32 or 36 communicates the control signaling to establish a call session, while a real-time link is established directly between the network telephone 30 or 34 and the remote system 100 for communicating voice or other types of audio data. In one embodiment, the call control messaging between the call control system and remote system, the control messaging between the call control system and the network telephone, and the call session between the network telephone and the remote system all occur over the data network 11.

The call control system 32 or 36 is also equipped with speech processing elements to allow it to communicate voice data with other devices on the data network 11. Thus, a user at the call control system 32 or 36 may select whether to use the call control system or the network telephone as the terminal device in the established call session. In addition, if the call control system 32 or 36 is powered off, the network telephone 30 or 34 may be used as a stand-alone device to communicate voice in call sessions over the data network 11.

Referring to FIG. 2, the components in the network telephone 30 or 34 and in the call control system 32 or 36 are illustrated in greater detail. The network telephone 30 or 34 includes a network interface 102 that is coupled to the data network 11. Above the network interface 102 are several software layers, including a device driver layer 104, a TCP/IP or UDP/IP stack 106, and an RTP layer 108. TCP is described in RFC 793, entitled "Transmission Control Protocol," dated September 1981; and UDP is described in RFC 768, entitled "User Datagram Protocol," dated August 1980. TCP and UDP are transport layers for managing connections between network elements over an IP network. Packets received by the network interface 102 are passed up through the several layers 104, 106 and 108. Control packets are transmitted by the TCP/IP or UDP/IP stack 106 to one or more control tasks 110 in the network telephone 30 or 34. The one or more control tasks 110 may be implemented as software routines executable on a control unit 1112. Instructions and data associated with the control tasks 110 may be stored in a storage device 114. The control tasks 110 are responsible for generation of control signaling as well as exchanging commands and responses with its associated call control system 32 or 36 over the data network 11.

Voice data may be passed through the RTP layer 108 to a speech processing application 116, which may also be executable on the control unit 112. For faster processing of voice data, a digital signal processor (DSP) 118 is included in the network telephone 30 or 34 to provide data intensive signal processing tasks. For example, the coder/encoder (CODEC) may be implemented in the DSP 118. The network telephone may also include a display screen to display text data associated with a call session. The size of the display screen 120 may be limited so that only limited amounts of text data may be displayed in the display screen 120. The network telephone also includes numerals buttons that may be controlled by button control circuitry 122. The buttons may include numeric buttons, speed dial buttons, a transfer button, a hold button, a redial button, and other telephony buttons. Activation of any one of the buttons may cause generation of some type of an indication (such as an interrupt) that is forwarded to the control tasks 110.

The call control system 32 or 36 also includes a network interface 150. Above the network interface 150 are several layers, including a device driver layer 152, a TCP/IP or UDP/IP stack 154, a SIP stack 156, and an RTP layer 158. The SIP stack 156 is responsible for processing or generating SIP requests and responses communicated over the data network 11. The SIP stack 156 is in communication with one or more control tasks 160 in the call control system 32 or 36. The SIP stack 156 is generally a state machine that provides parsing, processing, and generation of SIP requests and responses.

The call control tasks 160 are responsible for generating control signaling to establish call sessions over the data network 11 as well as to respond to received control signaling. In addition, the control tasks 160 are responsible for exchanging commands and responses with the network telephone 30 or 34 to establish such call sessions. The call control system 32 or 36 may also include one or more graphical user interface (GUI) routines 162 that control the presentation of information (text or graphical) on a display 164 of the call control system. Further, the user interface provided by the GUI routines 162 may include selectors for call control and indicators of the status of a call session.

In the illustrated arrangement, the RTP layer 158 sends audio data to, or receives audio data from, a CODEC 166. The CODEC 166 encodes or decodes voice data. A speech processing routine 168 may perform further processing of voice data. In further embodiments, the audio CODEC 166

and the speech processing routine **118** may be omitted. The various software routines in the call control system **32** or **36**, including the various layers **152**, **154**, **156**, and **158** as well as the control tasks **160**, CODECs **166**, speech processing routine **168**, and GUI routine **162**, are executable on a control unit **170**. The control unit **170** is coupled to a storage device **172** in which instructions and data associated with the various software routines may be stored.

In the illustrated example arrangement, to provide a voice or audio user interface to a user sitting at the call control system **32** or **36**, a peripheral controller **174** is coupled to a microphone **176** and a speaker or head phone **178** through which a user can talk or listen during a call session. If the call control system **32** or **36** is not speech-enabled, the microphone **176** and speaker or head phone **178** may be omitted.

One call control system **32** or **36** may be associated with a corresponding network telephone **30** or **34**. Thus, the network telephone **30** or **34** can identify which device is its controller. Similarly, a call control system **32** or **36** can identify the network telephone it is controlling. The network telephone **30** or **34** includes one or more fields **120** in the storage device **114** to store an identifier of its call controller, in this case the call control system **32** or **36**. The identifier may be in the form of a network address and port number. For example, an IP address and a TCP or UDP port may form part of the identifier of the call controller **120**. Similarly, the call control system **32** or **36** stores one or more fields **180** in the storage device **172** that stores the identifier of the network telephone it is controlling. Again, the identifier **180** may be in the form of a network address and port number, such as an IP address and a TCP or UDP port number. The identifier stored in the field **120** of the network telephone may be changed by a user to change the associated call control system. Similarly, the identifier stored in the field **180** of the call control system may be modified to change the controlled network telephone.

In further embodiments, one call control system may be associated with plural network telephones. Also, a single network telephone may be associated with plural call control systems.

The various control units in the network telephone **30** or **34**, the call control **32** or **36**, and any other system or device on the data network **11** may each include a microprocessor, a microcontroller, a processor card (including one or more microprocessors or controllers), or other control or computing devices. The storage devices referred to in this discussion may include one or more machine-readable storage media for storing data and instructions. The storage media may include different forms of memory including semiconductor memory devices such as dynamic or static random access memories (DRAMs or SRAMs), erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs) and flash memories; magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; and optical media such as compact disks (CDs) or digital video disks (DVDs). Instructions that make up the various software routines, modules, or layers in the various network elements may be stored in respective storage devices. The instructions when executed by a respective control unit cause the corresponding network element to perform programmed acts.

The instructions of the software routines, modules or layers may be loaded or transported to the network element in one of many different ways. For example, code segments including instructions stored on floppy disks, CD or DVD media, a hard disk, or transported through a network interface card, modem, or other interface device may be loaded into the system and executed as corresponding software routines, modules, or layers. In the loading or transport process, data signals that are embodied in carrier waves (transmitted over telephone lines, network lines, wireless links, cables, and the like) may communicate the code segments, including instructions, to the network element. Such carrier waves may be in the form of electrical, optical, acoustical, electromagnetic, or other types of signals.

Referring to FIG. **4**, in accordance with one embodiment, a screen **200** that may be provided by the control tasks **160** and graphical user interface routines **162** in the call control system **32** or **36** is illustrated. The screen **200** as shown in FIG. **4** includes various icons and items (generally referred to as indicators) to allow a user sitting at the call control system to initiate, terminate, and screen calls over the data network **11**. In the example shown in FIG. **4**, the screen **200** includes a menu **202**, a series of control buttons **204**, and a list **206** of potential callees. The list **206** provides the first and last names of potential callees as well associated electronic mail addresses (or other information such as telephone numbers and so forth). As illustrated in FIG. **4**, the name R. Smith may be highlighted in the list **206**. The address of R. Smith is displayed in an address field **208**. The address field **208** may include various formats, such as a PSTN number (e.g., 972-555-1234); a PSTN number and a proxy address (e.g., 972-555-1234@CTEXI300); an IP address (e.g., 47.161.18.72); a SIP address (e.g., rsmith@nortelnetworks.com); or a SIP address at a specific IP address (e.g., rsmith@47.161.18.72). Identifiers according to other formats may be illustrated in the address field **208** in further embodiments.

A status field **212** may also be included in the screen **200**, which may show the status as "not in call," "outgoing call to R. Smith," "incoming call from R. Smith," and so forth. A plurality of indicators **214** may also be provided in the screen **200**. A C indicator flashes when an incoming call has been missed. An S indicator gives an indication that call screening is active. A P indicator gives an indication that a SIP proxy is in use or not in use. An E indicator gives an indication of the state of the associated network telephone. Thus, the E indicator is at a first state if the network telephone is not active and at a second state if the network telephone is active and available. The E indicator may also be at a third state to indicate that a call is currently in progress.

The screen **200** is also capable of providing a pop-up menu **210** to allow a user to select one or several methods of contacting the desired callee. For example, a first option in the pop-up menu **210** is to call R. Smith. Another option is to send an electronic mail to R. Smith. A third option is to go to R. Smith's web site.

Other call control operations that may be performed by a user through the screen **200** includes volume control, screening of incoming calls, termination of a call session, and other operations.

Referring to FIG. **5**, once a call is established with either a caller or a callee, another screen **300** may be shown. A picture of the caller or callee may be displayed in the screen **300**. An icon **304** may be provided to allow the user to hang-up the call, and another icon **306** may be provided to allow the call to be placed on hold. A status field **308** indicates the current status of the call.

Referring to FIG. **6**, a message flow between a network telephone, a call control system, and a remote system is

illustrated. According to SIP, messages that may be exchanged between network elements include requests and responses. The remote system may be another call control system, one of the SIP client systems **38** and **40**, the data network-PSTN gateway **20**, or any other system capable of establishing a call session on the data network **11**. The remote system first sends (at **402**) an Invite request (according to SIP) to the call control system. The Invite request indicates that the receiving node is being invited to participate in a session. The message body of the Invite request contains a description (e.g., in SDP format) of the session to which the receiving node is being invited.

The call control system may then send (at **404**) a Ringing (SIP) response back to the remote system. The Ringing response indicates that the called user agent has located a possible location where the user has registered recently and is trying to alert the user. The call control system may then send (at **406**) a Connection_Req message to the network telephone to initiate a connection between the call control system and the network telephone. The messaging format between the network telephone and the call control system may be any predetermined format that allows call establishment and control to be performed by the call control system with the network telephone. One such format is the Unified Networks IP Stimulus Protocol, Draft Version 2.1, dated Dec. 7, 1999. In further embodiments, other interface protocols may be employed. A description of one embodiment of a protocol for message exchange between the network telephone and the call control system is provided in U.S. patent application Ser. No. 09/307,356, entitled "Telephony and Data Network Services at a Telephone," filed on May 7, 1999, which is hereby incorporated by reference.

The Connection_Req message is a generic message which includes one or more commands that indicates a request to establish a connection. The Connection_Req message may actually include a ring command to activate the ringer of the network telephone and other commands to activate the network telephone, such as activation of the handset, headset, microphone, speaker, and so forth. The network telephone may then send back (at **408**) an Ack_Req message to the call control system to acknowledge that the network telephone is available and ready. The Ack_Req message may also be a generic message to acknowledge receipt of the Connection_Req message. Upon receipt of Ack_Req message from the network telephone, the call control system sends (at **410**) a 200 OK SIP response to the remote system to indicate that the request has succeeded. The remote system then sends (at **412**) an Ack request (according to SIP) to the call control system. The Ack request confirms that the client has received a final response to an Invite request.

Upon receipt of Ack request, the call control system sends (at **414**) a Remote_Answer message to the network telephone to indicate a request to establish a path for a call session. If accepted, the network telephone then sends (at **416**) an Ack_Answer message back to the call control system. The Remote_Answer message may be a generic message that includes one or more commands to activate the network telephone for call session. One such command is a command to open or connect the audio stream to the handset, headset, microphone and speaker of the network telephone. At that point, a voice path is established (at **418**) directly between the network telephone and the remote system. The voice path may be an RTP link over the data network **11**.

To terminate the call, the remote system may issue (at **420**) a Bye request to the call control system. The call control system then responds (at **422**) with a 200 OK, indicating that the call has been terminated. Then, the call control system sends (at **424**) a Disconnect_Req message to the network telephone to disconnect the network telephone from the data network. The Disconnect_Req message be a generic message including one or more commands to deactivate various components of the network telephone. For example, the audio stream may be closed or disconnected, and the handset, headset, microphone, and speaker may be deactivated. The network telephone then returns (at **426**) an Ack_Disconnect message back to the call control system to indicate that the call has been disconnected.

Referring to FIG. **7**, an outgoing call message flow is illustrated. In the illustrated example, the user can initiate the call from the call control system. However, the user can also make the external call from the network telephone by entering the desired number in appropriate buttons of the network telephone. In that case, messages are exchanged between the network telephone and the call control system initially to indicate to the call control system that the user has started a phone call from the network telephone.

To start the call session, the call control system sends (at **502**) an Invite request to the remote system. The remote system then sends back (at **504**) a Ringing response. In response, the call control system sends (at **506**) a Remote_Alerting message to the network telephone indicating that the call has been placed. The network telephone then returns (at **508**) an Ack_Alerting message. At some point, the remote system, once it has answered the call, issues (at **510**) a 200 OK message to the call control system. In response, the call control system then sends (at **512**) an Ack request back to the remote system. The call control system also sends (at **514**) a Remote_Answer message to the network telephone, which returns (at **516**) an Ack_Answer message to the call control system. At that point, a voice path (e.g., an RTP path) is established (at **518**) between the network telephone and the remote system over the data network **11**.

To terminate the call, the remote system may issue (at **520**) a Bye request. In response, the call control system may terminate the call by sending (at **522**) a 200 OK message. The call control system then sends (at **524**) a Disconnect_Req message to the network telephone, which returns (at **526**) an Ack_Disconnect message to the call control system. At this point, the RTP voice path is terminated.

While the invention has been disclosed with respect to a limited number of embodiments, those skilled in the art will appreciate numerous modifications and variations therefrom. It is intended that the appended claims cover all such modifications and variations as fall within the true spirit and scope of the invention.

What is claimed is:

1. A method of communicating over a data network, comprising:

providing a user interface in a control system for establishing call sessions;

communicating, by the control system, one or more control messages over the data network to establish a call session with a remote device in response to receipt of a request through the user interface; and

transmitting one or more commands to a voice device connected to the data network and associated with the control system to establish the call session between the voice device and the remote device over the data network.

2. The method of claim 1, wherein the communicated one or more control messages and the transmitted one or more commands are according to different formats.

3. The method of claim 1, wherein transmitting the one or more commands to the voice device includes transmitting one or more commands to a network telephone including a network interface to the data network.

4. The method of claim 1, wherein establishing the call session includes establishing a Real-Time Protocol session over the data network.

5. The method of claim 1, wherein communicating the one or more control messages includes communicating messages according to a protocol defining real-time, interactive call sessions over a packet-switched data network.

6. The method of claim 1, wherein communicating the one or more control messages includes communicating one or more Session Initiation Protocol messages.

7. The method of claim 1, further comprising storing, in the control system, an identifier of the voice device.

8. The method of claim 7, wherein storing the identifier includes storing an Internet Protocol address and a port of the voice device.

9. The method of claim 1, further comprising receiving an indication from the voice device to establish another call session with the remote device.

10. The method of claim 1, further comprising displaying graphical user interface information of the call session on the control system.

11. The method of claim 1, further comprising terminating the call session using either the user interface or the voice device.

12. A method of communicating over a data network, comprising:

in a control system, communicating one or more control messages over the data network to establish a call session with a remote device coupled to the data network;

transmitting one or more commands to a voice device coupled to the data network;

establishing the call session between the voice device and the remote device over the data network; and

displaying information associated with the call session on the control system.

13. The method of claim 12, wherein displaying the information includes displaying graphical user interface information.

14. The method of claim 12, wherein communicating the one or more control messages includes communicating Session Initiation Protocol messages.

15. The method of claim 12, further comprising providing one or more indicators for call control in the control system.

16. The method of claim 12, further comprising communicating Real-Time Protocol messages between the voice device and the remote device over the data network.

17. The method of claim 12, further comprising identifying, in the control system, an address of the voice device to be controlled by the control system.

18. The method of claim 12, further comprising providing a user interface on a display of the control system, the user interface enabling selection of one or more criteria associated with the voice device.

19. The method of claim 18, wherein the one or more criteria includes selection of the voice device for use in a voice session established by the control system.

20. The method of claim 19, wherein the one or more criteria includes an identifier of the voice device.

21. The method of claim 12, further comprising providing voice processing components in the control system and selecting one of the voice processing components and the voice device to communicate in the established call session.

22. The method of claim 21, further comprising receiving user selections entered in a user interface of the control system to select one of the voice processing components and the voice device.

23. The method of claim 21, further comprising redirecting selection to the other one of the voice processing components and voice device.

24. The method of claim 12, wherein the data network includes a packet-switched data network.

25. A system for controlling a voice device connected to a data network, comprising:

a user interface including one or more selectors for call control relating to call sessions;

a controller adapted to receive a request from the user interface and to generate one or more messages for communication over the data network to establish a call session with a remote device; and

an interface to transmit one or more commands relating to the call session to the voice device to establish a link between the voice device and the remote device over the data network.

26. The system of claim 25, wherein the one or more messages include Session Initiation Protocol messages.

27. The system of claim 26, further comprising a module to process the one or more Session Initiation Protocol messages.

28. The system of claim 25, wherein the interface includes a network interface for coupling to the data network.

29. The system of claim 25, further comprising a storage element including an identifier of the voice device.

30. The system of claim 25, wherein the user interface includes one or more elements to display information relating to the call session.

31. The system of claim 30, wherein the information includes graphical information.

32. An article including one or more machine-readable storage media containing instructions for controlling voice communications over a data network, the instructions when executed causing a system to:

provide a user interface in the system to display information associated with a call session;

communicate one or more control messages over the data network with a remote device to establish the call session between a voice device and the remote device; and

control the voice device during the call session.

33. The article of claim 32, wherein the one or more storage media contain instructions that when executed cause the system to communicate Session Initiation Protocol messages.

34. The article of claim 32, wherein the one or more storage media contain instructions that when executed cause the system to display a picture of a callee.

35. The article of claim 32, wherein the one or more storage media contain instructions that when executed cause the system to display icons selectable by a user for call control.

36. A data signal embodied in a carrier wave comprising one or more code segments containing instructions for controlling a call session over a data network, the instructions when executed causing a system to:

provide a user interface in the system for establishing the call session;

communicate one or more control messages over the data network to establish the call session with a remote

device in response to a request received through the user interface; and

transmit one or more commands to a voice device connected to the data network and associated with the control system to establish the call session between the voice device and the remote device over the data network.

37. The method of claim **1**, further comprising receiving activation of an element in the user interface to indicate establishment of the call session between the voice device and the remote device.

38. The method of claim **13**, further comprising receiving activation of one or more selectors in the graphical user interface information for initiating the call session between the voice device and the remote device.

39. The system of claim **25**, wherein the user interface comprises a graphical user interface, and wherein the controller is adapted to receive activation of one or more selectors in the graphical user interface to establish the call session.

* * * * *

EXHIBIT I

US006337858B1

(12) **United States Patent**
Petty et al.

(10) **Patent No.: US 6,337,858 B1**
(45) **Date of Patent: Jan. 8, 2002**

(54) **METHOD AND APPARATUS FOR ORIGINATING VOICE CALLS FROM A DATA NETWORK**

(75) Inventors: **Douglas Petty**; **Michael William Petras**, both of Nepean; **Raul San Martin**, Ottawa; **Alan Stuart Frank**, Nepean; **Vivek Kapil**, Ottawa; **Dhansukh Mistry**, Kanata, all of (CA)

(73) Assignee: **Nortel Networks Limited**, Montreal (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **08/948,975**

(22) Filed: **Oct. 10, 1997**

(51) **Int. Cl.**[7] **H04L 12/56**
(52) **U.S. Cl.** **370/356**; 370/410
(58) **Field of Search** 370/352, 353, 370/354, 355, 356, 400, 401, 466, 402, 403, 404, 405, 410; 379/201, 209, 211, 212, 214, 215, 88.17, 93.01, 93.09; 709/704, 206, 227, 249

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,191,860 A		3/1980	Weber	179/18 B
5,602,846 A	*	2/1997	Holmquist et al.	370/384
5,675,507 A	*	10/1997	Bobo, II	709/206
5,742,596 A	*	4/1998	Baratz et al.	370/356
5,781,614 A	*	7/1998	Brunson	379/88.14
5,805,587 A	*	9/1998	Norris et al.	370/352
5,862,203 A	*	1/1999	Wulkan et al.	379/114
5,870,549 A	*	2/1999	Bobo, II	709/206
5,884,032 A	*	3/1999	Bateman et al.	709/204
5,970,126 A	*	10/1999	Bowater et al.	379/114
5,974,043 A	*	10/1999	Solomon	370/352
5,991,291 A	*	11/1999	Asai et al.	370/352
6,069,890 A	*	5/2000	White et al.	370/352

FOREIGN PATENT DOCUMENTS

FR	2677517 A1		10/1991	H04M/11/00
WO	97/14238	*	4/1997	H04L/12/46
WO	97/19548	*	5/1997	H04M/15/00

OTHER PUBLICATIONS

"Internet Call Center", Lucent Technologies web page, 1996.
"Internet Call Center", Lucent Technologies, 1996.
News Release "Lucent Technologies debuts WebCall in Japan", Lucent Technologies, May 8, 1997.
News Release "Lucent Technologies brings award–winning advanced call center capabilities to small businesses", Lucent Technologies, Jun. 10, 1997.
"Lucent Technologies ' Internet Telephony Server", Lucent Technologies, 1997.

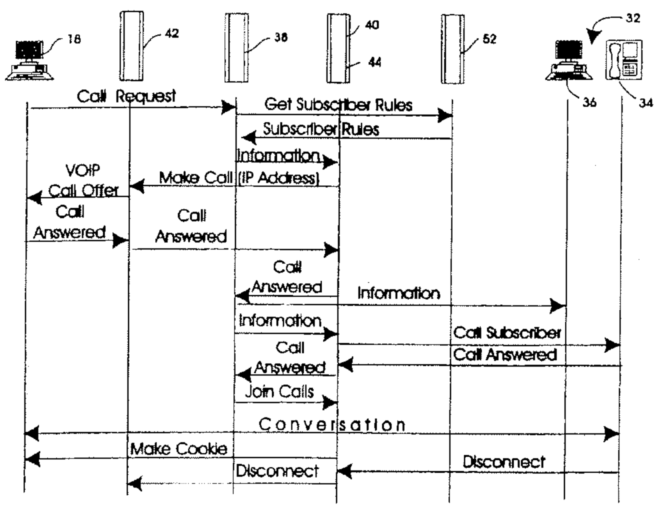
* cited by examiner

Primary Examiner—Kwang B. Yao
(74) *Attorney, Agent, or Firm*—Max R. Wood; Swabey Ogilvy Renault

(57) **ABSTRACT**

A method and apparatus for providing voice communications between two parties using computer controlled telephony hardware which is separate from the PSTN is described. The voice communications may be voice over Internet or PSTN voice connections, or any combination of the two. The apparatus includes a WEB server, a computer telephony server, a Voice over IP gateway and an operations, administration and maintenance server. The apparatus is suitably installed and operated by a service provider such as an Internet Service Provider. The advantage is unexcelled control over call setup, transfer and conference functions. A further advantage is the rapid, unfettered development of new services without compatibility issues with the PSTN.

30 Claims, 10 Drawing Sheets



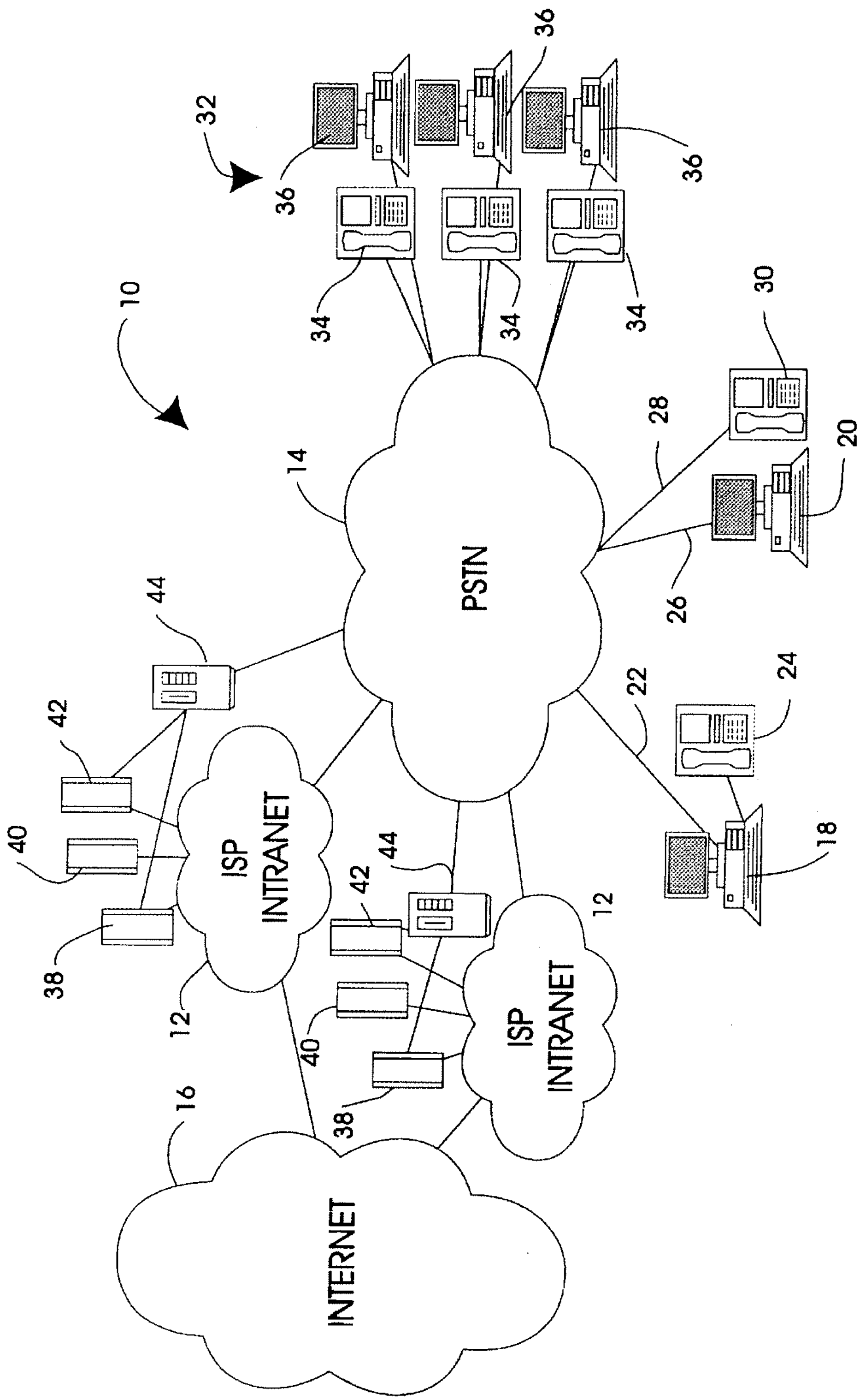

PSTN
ISP INTRANET
ISP INTRANET
INTERNET
10
12
12
14
16
18
20
22
24
26
28
30
32
34
34
34
36
36
36
38
38
40
40
42
42
44
44

FIG. 1

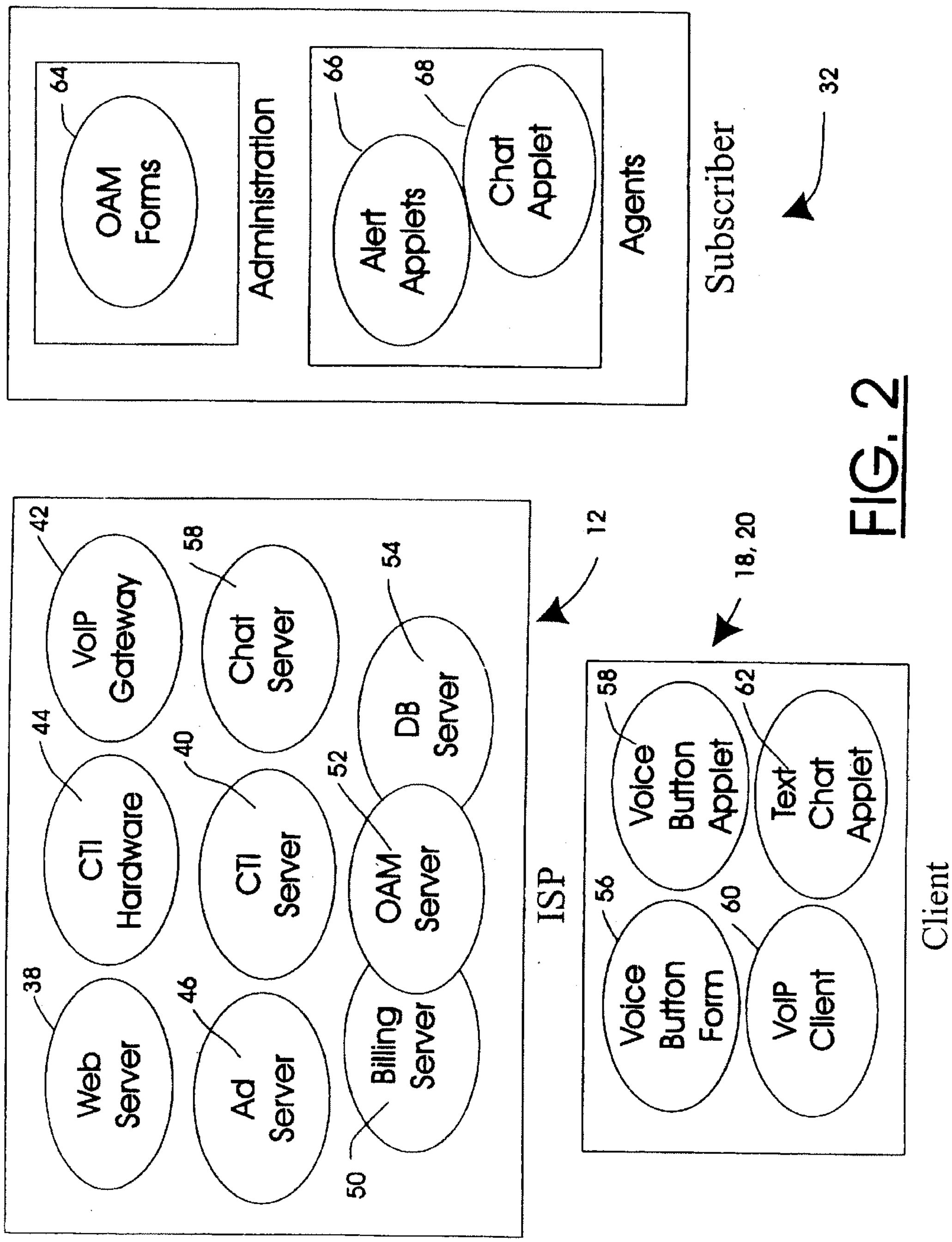

OAM Forms
64
Administration
Alert Applets
66
Chat Applet
68
Agents
Subscriber
32
Web Server
38
CTI Hardware
44
VoIP Gateway
42
Ad Server
46
CTI Server
40
Chat Server
58
Billing Server
50
OAM Server
52
DB Server
54
ISP
12
Voice Button Form
56
Voice Button Applet
58
VoIP Client
60
Text Chat Applet
62
Client
18, 20

FIG. 2

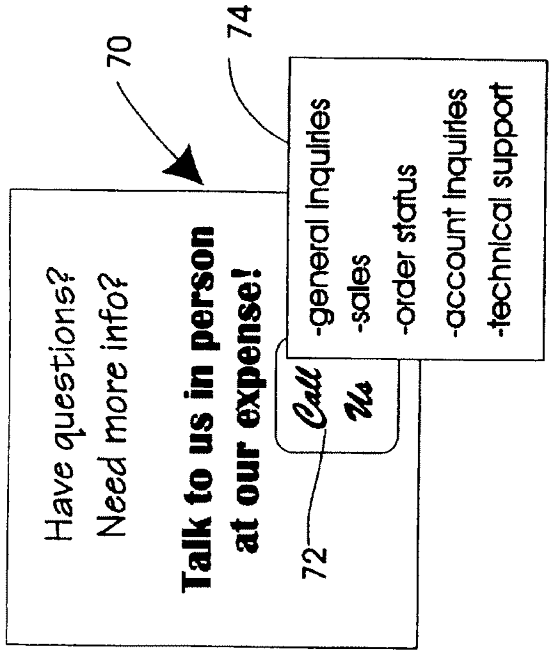


FIG. 3b

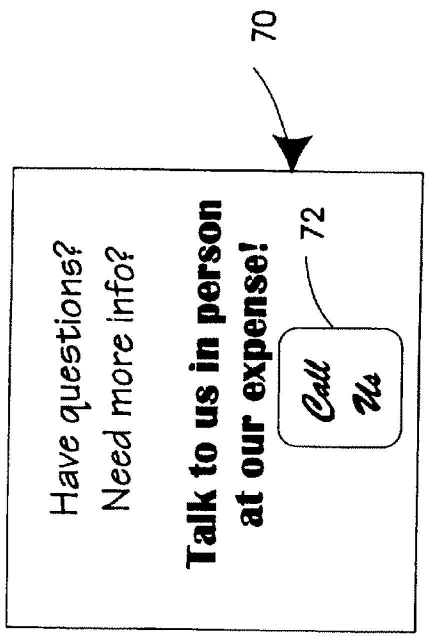


FIG. 3a

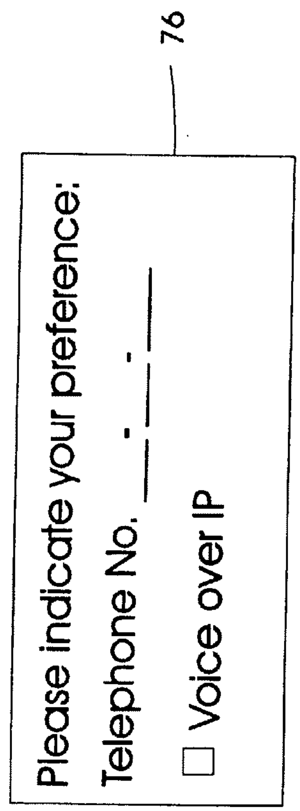


FIG. 4

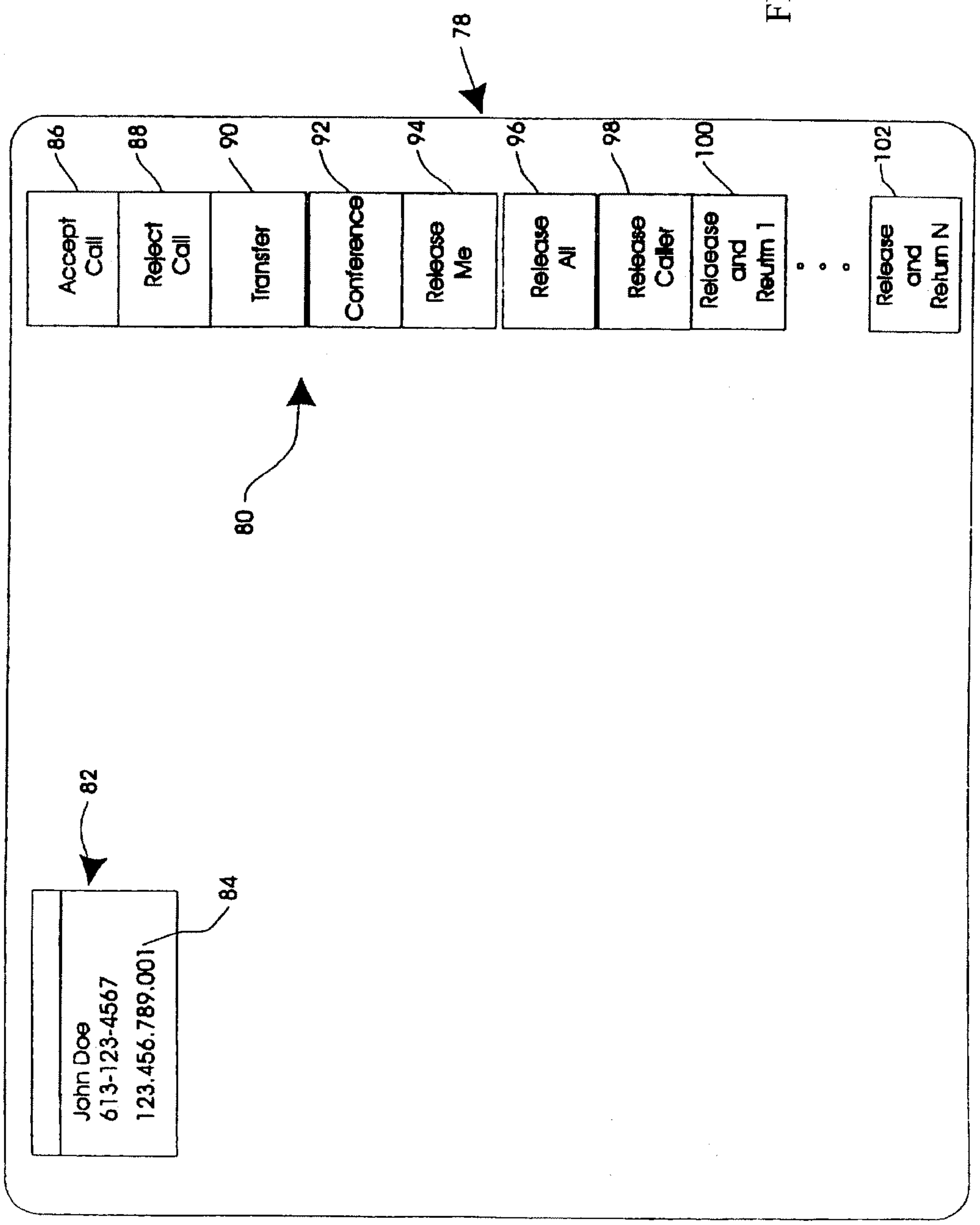

78
86
Accept Call
88
Reject Call
90
Transfer
92
Conference
94
Release Me
96
Release All
98
Release Caller
100
Release and Return 1
102
Release and Return N
80
82
John Doe
613-123-4567
123.456.789.001
84

FIG. 5

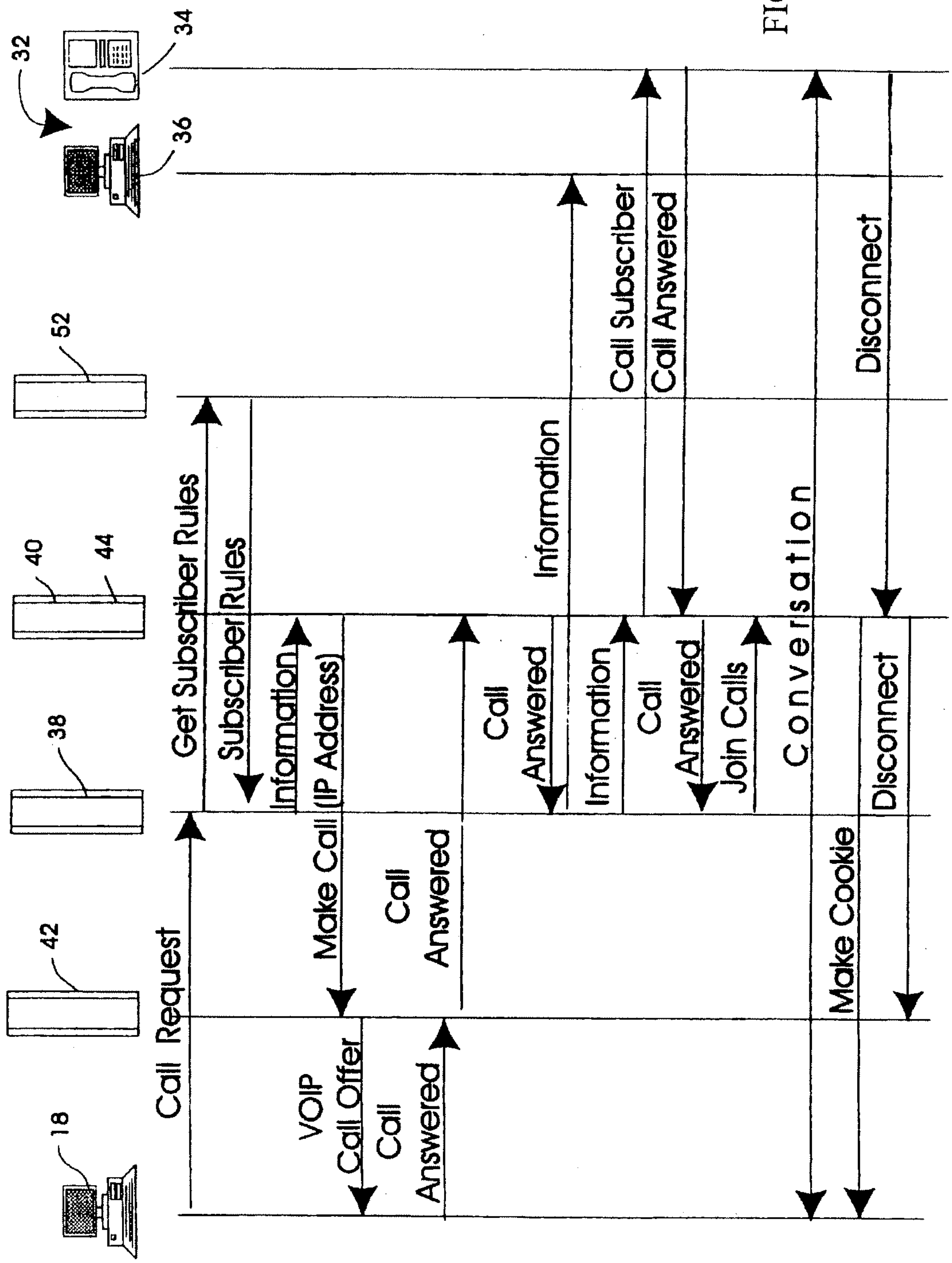

32
34
36
52
40
44
38
42
18
Call Request
Get Subscriber Rules
Subscriber Rules
Information
Make Call (IP Address)
VOIP Call Offer
Call Answered
Call Answered
Call Answered
Information
Information
Call Subscriber
Call Answered
Call Answered
Join Calls
Conversation
Make Cookie
Disconnect
Disconnect

FIG. 6

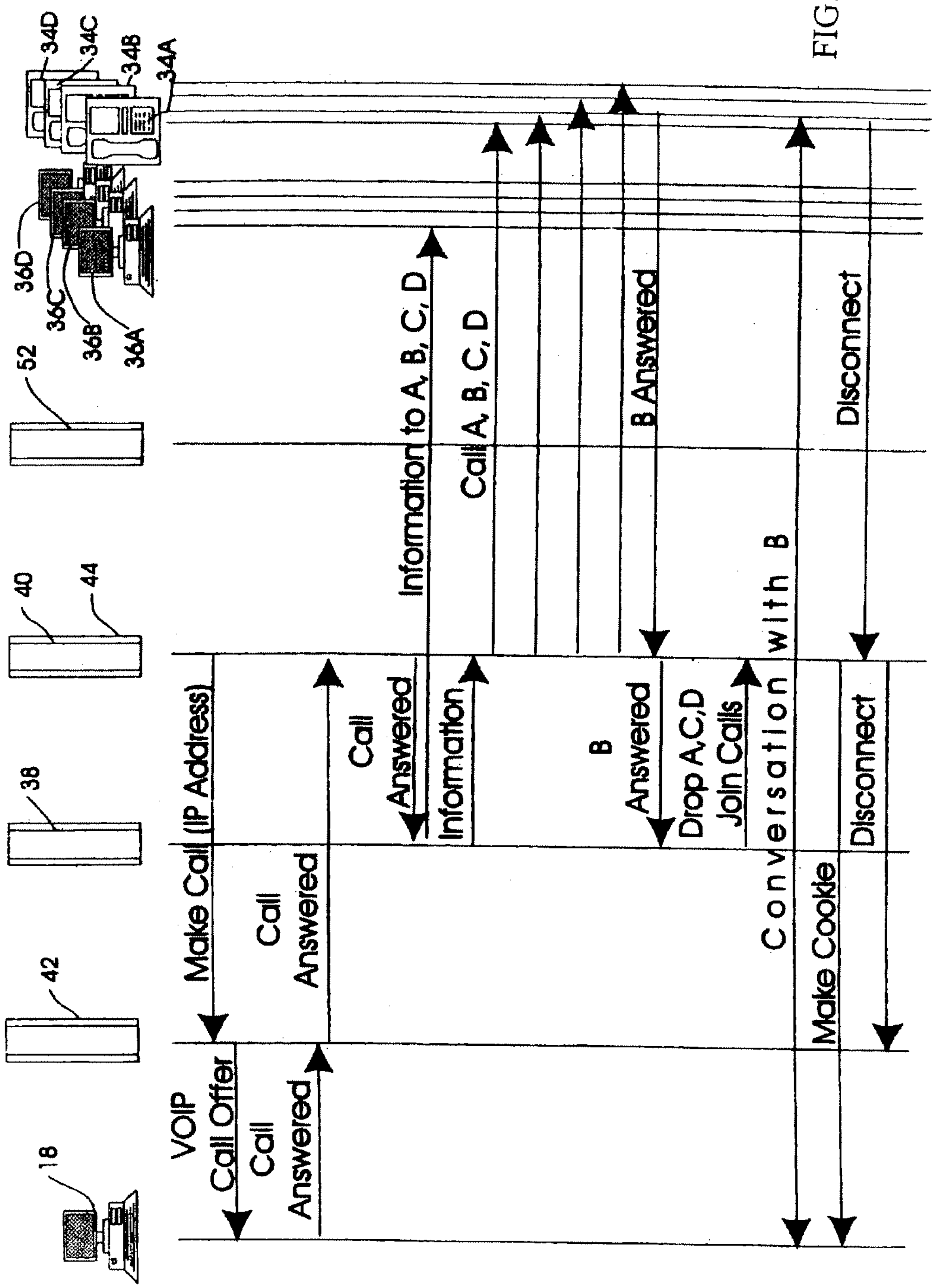
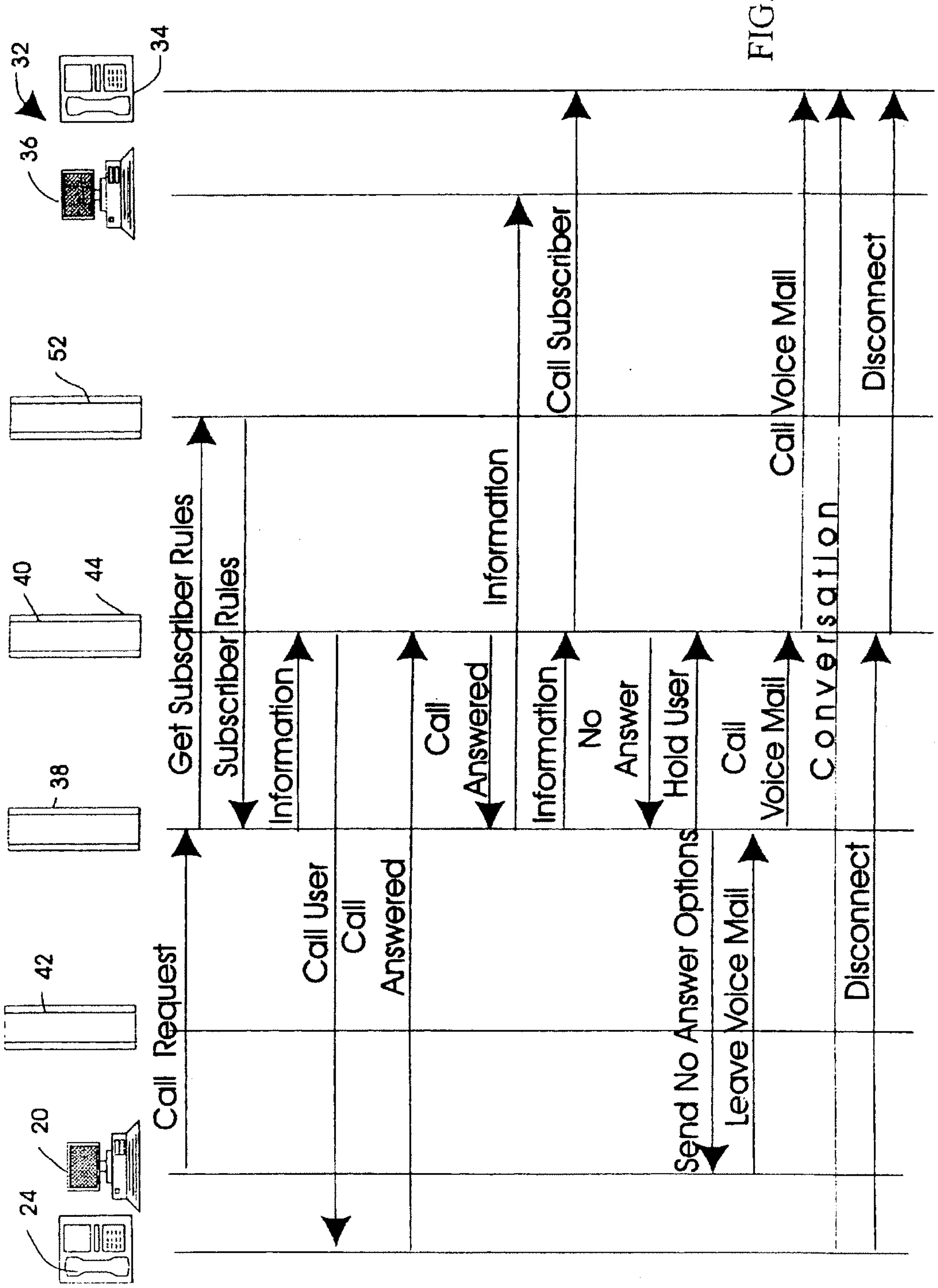

34D
34C
34B
34A
36D
36C
36B
36A
52
40
44
38
42
18
Information to A, B, C, D
Call A, B, C, D
B Answered
Disconnect
Conversation with B
Make Call (IP Address)
Call Answered
Call Answered
Information
B Answered
Drop A,C,D Join Calls
Disconnect
Make Cookie
Disconnect
VOIP Call Offer
Call Answered

FIG. 7



32
34
36
52
40
44
38
42
20
24
Call Request
Get Subscriber Rules
Subscriber Rules
Information
Call User
Call Answered
Call Answered
Information
Call Subscriber
Information
No Answer
Hold User
Send No Answer Options
Leave Voice Mail
Call Voice Mail
Call Voice Mail
Conversation
Disconnect
Disconnect

FIG. 8

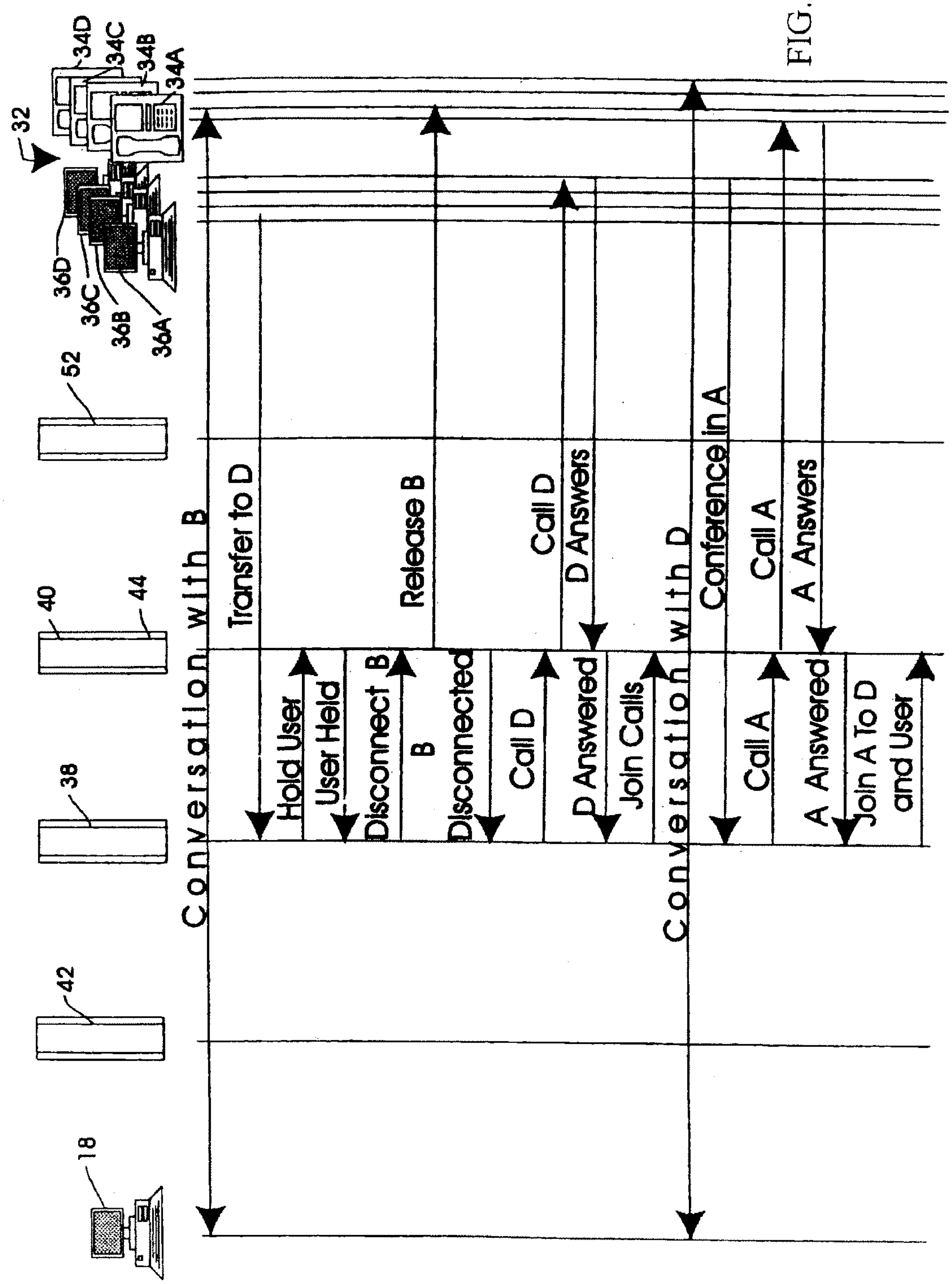
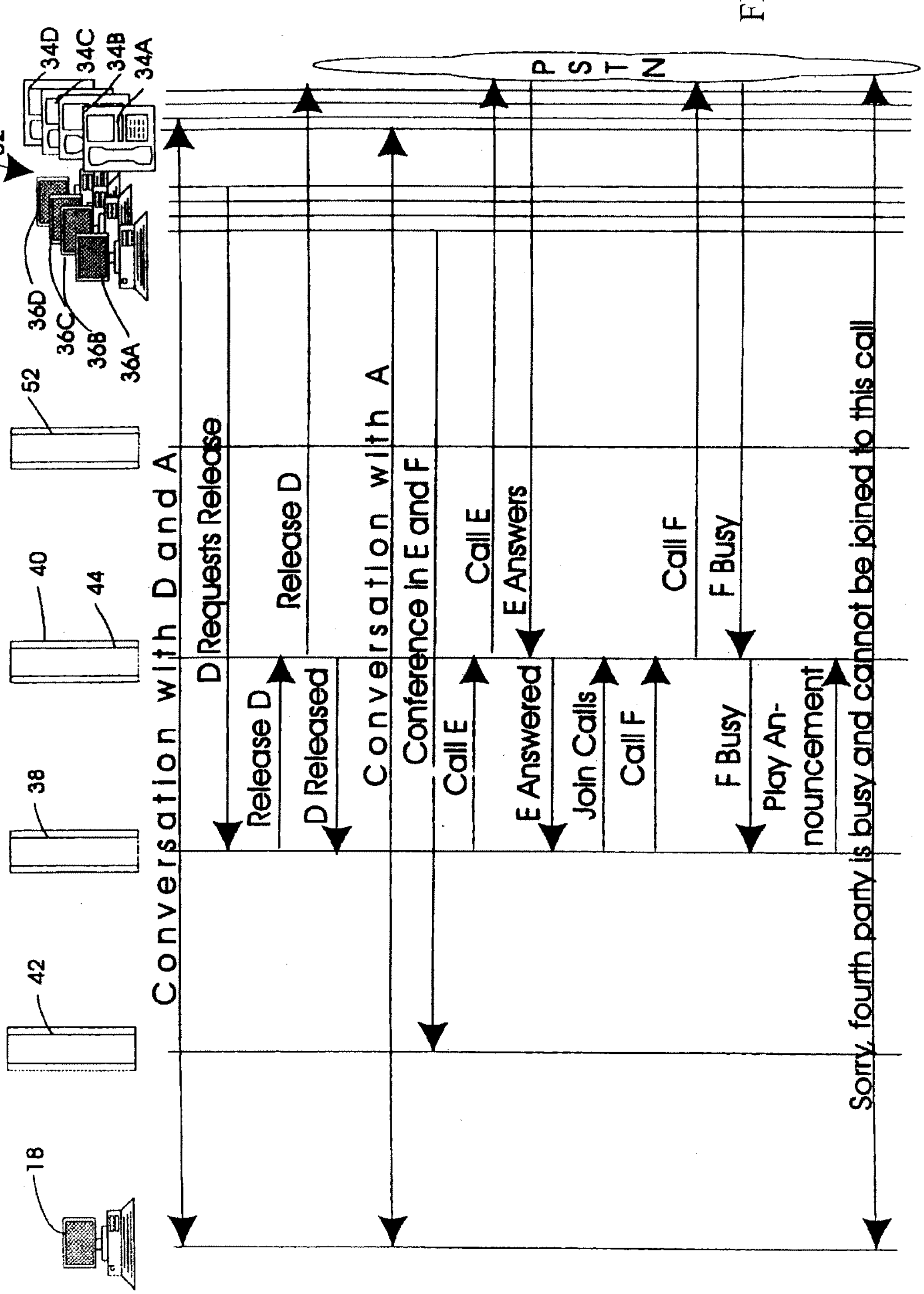

32
34D
34C
34B
34A
36D
36C
36B
36A
52
40
44
38
42
18
Conversation with B
Transfer to D
Hold User
User Held
Disconnect B
Release B
B Disconnected
Call D
Call D
D Answers
D Answered
Join Calls
Conversation with D
Conference in A
Call A
Call A
A Answers
A Answered
Join A To D and User

FIG. 9A

FIG. 9B



18
42
38
40
44
52
36A
36B
36C
36D
32
34A
34B
34C
34D
PSTN
Conversation with D and A
D Requests Release
Release D
Release D
D Released
Conversation with A
Conference In E and F
Call E
Call E
E Answers
E Answered
Join Calls
Call F
Call F
F Busy
F Busy
Play An-nouncement
Sorry, fourth party is busy and cannot be joined to this call

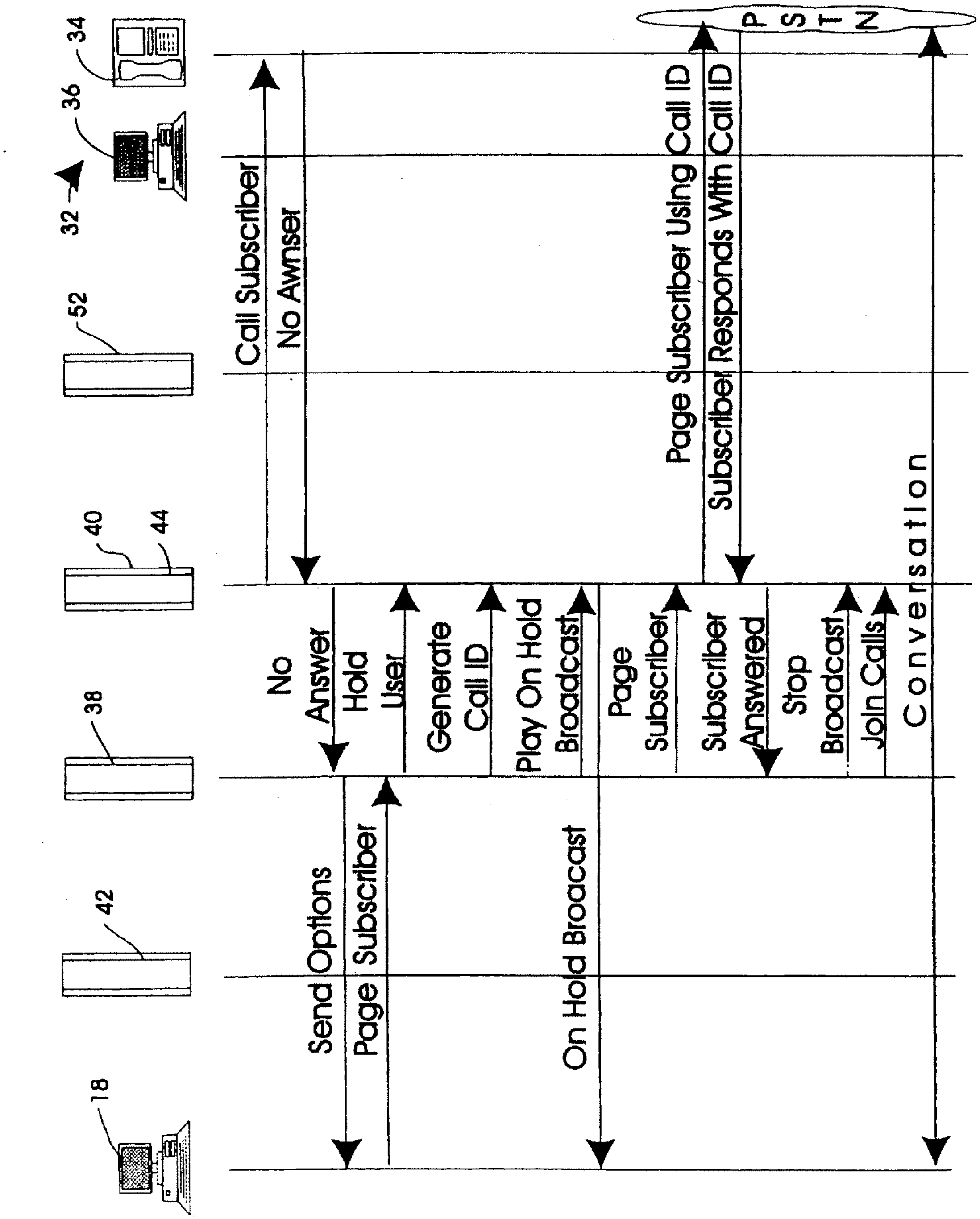

34
36
32
52
40
44
38
42
18
Call Subscriber
No Awnser
Page Subscriber Using Call ID
Subscriber Responds With Call ID
PSTN
No Answer
Hold User
Generate Call ID
Play On Hold Broadcast
Page Subscriber
Subscriber Answered
Stop Broadcast
Join Calls
Conversation
Send Options
Page Subscriber
On Hold Broadcast

FIG. 10

METHOD AND APPARATUS FOR ORIGINATING VOICE CALLS FROM A DATA NETWORK

FIELD OF THE INVENTION

This invention relates to voice communications and, in particular, to providing voice communications between two parties using computer controlled telephony hardware which is outside the Public Switched Telephone Network.

BACKGROUND OF THE INVENTION

The importance of effective communication tools for use by business in advertising, product support, promotion and sales has long been recognized. Much inventive ingenuity has been invested in the development of tools for facilitating and improving business communications. A prime example is modern toll free calling services, commonly referred to as 800 calling services, which were enabled by an invention described in U.S. Pat. No. 4,191,860 to Weber. 800 calling services have since been vastly improved and made increasingly available as a business tool. In particular, sophisticated call centers have been developed to permit persons generally described as agents to field calls from the public for the purposes of disseminating information, supporting products and selling products.

In recent years, the Internet has also become an important and increasingly accepted tool for disseminating business information and promoting products as well as providing support to business customers. Use of the Internet and, in particular, the Worldwide Web has become pervasive in the industrialized world. Consequently, the importance of the Worldwide Web as a business communications tool has been recognized. While its importance is recognized, the electronic information provided by many companies on the Worldwide Web is static and designed to appeal to the widest audience. It is therefore neither adequate nor designed to respond to the specific needs of individuals. In order to obtain the breadth of information most individuals require, it is generally necessary to call the business and verbally request specific information. This process is inefficient because the caller must explain to the business contact what information was viewed, as well as what further particulars are required. This inefficiency has been recognized as a drawback for some time. Consequently, facilities have been developed to permit individuals browsing the Worldwide Web to place voice calls from Web pages. Such facilities are commonly called "voice buttons". One such facility is described, for example, in applicant's co-pending U.S. patent application Ser. No. 08/652,659 filed on May 28, 1996 and entitled METHODS AND APPARATUS FOR ORIGINATING VOICE CALLS. That application describes methods and apparatus for originating voice calls between voice terminals using a data terminal and a data service node without requiring interaction with human attendants or interactive voice response systems to complete the call. In accordance with the method, an apparatus connected with a data network is used to send commands to a PSTN switching node to initiate a call between the voice terminals to permit the person browsing the Web to readily communicate with the business which owns the Web page, assuming that the user has a voice terminal available for accepting the call.

Other apparatus have also been developed, for example, Lucent Technologies® have developed methods and apparatus for integrating Worldwide Web pages and 800 call centers to permit browsers of a Worldwide Web page to launch a Voice over IP or a voice terminal connection to an

800 call center while providing an agent at the call center receiving the call with a data terminal screen "pop" of the same page the browser is observing. The agent is therefore visually attuned to the browser's position and enabled to quickly and efficiently answer questions or close transactions.

While such facilities greatly enhance a business' ability to respond to public need, they do not provide a viable solution for all businesses or provide a full range of potential services that can be developed through the inventive use of computer telephony integration. It is widely recognized that in the information age, small business is the driving force of the economy. While telecommunications service providers have recognized the need of small business to have access to such communications services as 800 call service and have lowered entry barriers to such services, 800 service is not necessarily well adapted to meet the needs of small business. This is particularly true where a small business derives most or all of its clientele from a local area. In that case, 800 service is of substantially no value. Even if a small business' clientele extends well beyond a local calling area, because of the inherent limitations in current 800 service offerings, many small businesses find it uneconomical or impractical to subscribe to an 800 service for a variety of reasons, including the general inflexibility of 800 call terminations which substantially require at least one full time person dedicated to servicing 800 inquiries.

Consequently, there is a recognized need for a more flexible yet cost effective method of providing voice services to medium and small sized businesses and, in particular, to providing voice services to medium and small sized businesses having a presence on the Worldwide Web.

SUMMARY OF THE INVENTION

It is an object of the invention to provide a method and apparatus to permit voice communications, between a consumer browsing the Internet and a business having an interactive information page on the Internet, independent of whether the user is within a toll free calling area of the business.

It is a further object to the invention to provide an apparatus and method to permit voice communications between the consumer and the business which is particularly adapted to serve small and medium sized businesses.

It is yet a further object to the invention to provide a method and apparatus for voice communications between the consumer and the business which employs a rule base to adapt the service to the resources of the business in order to permit maximum flexibility and ensure optimum service.

It is a further object to the invention to provide an apparatus and method adapted to offer call center functionality without call center infrastructure.

It is yet another object of the invention to provide an apparatus and method adapted to permit the rapid development of new call handling features for business subscribers having a presence on the Internet.

These and other objects are realized in a method of providing voice communications between two parties using computer controlled telephony hardware which is separate from the Public Switched Telephone Network (PSTN), at least one of the parties having access to a data network, comprising:

originating a first voice connection from the computer controlled telephony hardware in response to a call request received from the data network;

originating a second voice connection from the computer controlled telephony hardware in response to the call request received from the data network; and

bridging together the first and second voice connections at the computer controlled telephony hardware to permit voice communications between the two parties.

In accordance with a further aspect of the invention there is provided a method of providing voice communication between a service subscriber and a user of a data network accessing an interactive information page available on the network using a data terminal, the information page including a voice communication request button relating to the service subscriber, comprising the steps of:

a) accepting a voice communication request from the user;

b) analyzing the voice communication request or a preference file stored on the user's data terminal to determine a voice connection medium preferred by the user;

c) consulting a rule base using a code that identifies the subscriber to locate at least one rule which determines how a voice connection is to be established with the subscriber;

d) operating computer telephony hardware to place a first call to the user by the preferred voice connection medium;

e) operating the computer telephony hardware to place a second call to the subscriber in accordance with the at least one rule located in the rule base;

f) operating the computer telephony hardware to bridge the first and second calls;

g) monitoring the data network for an indication that a modification of a configuration of the voice communication is desired by either the user or the subscriber and changing the configuration of the voice communication if an indication for change is received; and

h) monitoring the first and second calls and disconnecting the other of the first and second calls when either of the first or second calls is disconnected.

In yet a further aspect to the invention, there is provided a system for providing voice communication between a service subscriber and a user of a data network accessing an interactive information page available on the network using a data terminal, comprising in combination:

at least one computing machine programmed with the functionality of:

a) a voice button server for accepting voice communication requests either directly or indirectly from a user accessing an interactive information page on a data network;

b) an operations, management and maintenance server to permit the establishment and maintenance of rule bases and related controls; and

c) a computer telephony server for controlling computer telephony hardware;

a voice over data network gateway for converting voice data packets to voice telephony format; and

computer telephony hardware which may be controlled by the computer telephony server to initiate voice connections on a switched telephone network or a voice gateway to a data network, to bridge calls and to respond to a set of predefined voice connection control commands which may originate from either the service subscriber or the user.

The invention therefore provides a method and apparatus for permitting a user browsing a data network such as the

Worldwide Web (WWW) or the Internet to establish voice communications with a service subscriber associated with an interactive information page of the data network. The interactive information page includes a "voice button" which may be activated by the user to initiate voice communications to obtain more detailed information about the items being advertised. The user may select a preferred medium for the voice communication. If the user connects to the data network with his phone line, the user is likely to prefer a "Voice over IP" connection with the service subscriber. If the user has a data line as well as a phone line, the user will likely prefer a voice terminal connection over the phone line due to the superior sound quality. When a first time user activates the voice button, a data message is sent directly or indirectly from the user's terminal to a voice button server which determines from the message the identity of the service subscriber and the user's preferred voice connection medium. For repeat users, the preferred call medium may be looked up using a preference file stored on the user's data terminal. Such preference files are commonly called "cookies". The voice button server also examines a rule base to determine how the service provider wishes his leg of the call to be established. All call initiation and bridging is effected from computer telephony hardware comprising a switch fabric and a switch controller which permits ultimate flexibility with respect to the establishment, bridging, transfer and other call operations. The computer telephony hardware is the originating point for each leg of a voice connection. When a user requests voice communications, at least two calls are placed. A first call over the preferred medium (phone or Voice over IP) to the user and a second call to an agent for the service subscriber. Once both calls are established they are bridged by the computer telephony hardware to provide the voice communication

Independent control of each leg of the call provides flexibility unrealized in prior art systems designed to enable voice communications between a browser of a data network and a service subscriber having a Web site on the data network. In addition, since only standard Public Switched Telephone Network (PSTN) functionality or standard Voice over IP functionality is used in call completion, no knowledge, control or adaptation of those protocols or procedures is required to construct, install or operate the system in accordance with the invention. Furthermore, there are no compatibility issues with the PSTN respecting deployment of the method or apparatus in accordance with the invention.

The method and apparatus in accordance with the invention are particularly adapted to provide efficient and flexible service from small to mid-sized businesses and to provide call center functionality without call center infrastructure. A rule base is used to permit the service to be configured to the specific needs of each service subscriber. Flexible features such as "simultaneous notification groups" (SNGs), "agent locator" and "automated page return" are just three examples of the flexibility enabled. In SNGs, two or more calls are launched simultaneously to different telephone numbers or IP addresses and the call which is answered first is bridged to the user requesting voice communications. With "agent locator", a plurality of calls may be launched simultaneously to different numbers or IP addresses for the same individual or different locations where the individual may be expected to be at any given time. A call is selected when an answering party enters a code identifying that party to be the agent sought. With "automated page return" a subscriber is paged using an identification number for a call being held. When the paged subscriber calls in and enters the identification number he is automatically bridged to the held calling party.

The apparatus in accordance with the invention preferably includes a voice button server, a voice over IP gateway, an operations, administration and maintenance server, and a computer telephony server which operates computer telephony hardware. Each server may be a separately programmed computing machine or the functionality of several servers may be configured in a single machine. The apparatus in accordance with the invention is particularly adapted for use by Internet Service Providers (ISPs) to permit the ISPs to offer their business subscribers a flexible service for business advertising, promotion, support and sales. The rule base permits each business to adapt the service to its own needs. It also permits the business to control call permissions and determine call completion patterns.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention will now be further explained by way of example only and with reference to the following drawings, wherein:

FIG. **1** is a schematic diagram of an overview of a communications network configured with apparatus in accordance with the invention;

FIG. **2** is a schematic diagram of the software and hardware components required to practise the invention;

FIGS. **3***a* and **3***b* are schematic diagrams representing voice buttons in accordance with a preferred embodiment of the invention;

FIG. **4** is a voice request preference form in accordance with a preferred embodiment of the invention;

FIG. **5** is a schematic diagram representing a business agent screen display in accordance with a preferred embodiment of the invention;

FIG. **6** is a schematic diagram of a typical basic call flow of a call process in accordance with the invention;

FIG. **7** is a schematic diagram illustrating one optional call process using apparatus and methods in accordance with the invention;

FIG. **8** is a schematic diagram illustrating the call flow of another optional call process using the methods and apparatus in accordance with the invention;

FIGS. **9***a* and **9***b* are schematic diagrams of portions of call flows showing the flexibility of call handling using the methods and apparatus in accordance with the invention; and

FIG. **10** is a schematic diagram showing a portion of a call flow for a further call feature enabled by the methods and apparatus of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The invention relates to novel methods and apparatus for providing a service to subscribers with a presence on the Worldwide Web or the Internet. The service provides the tools to permit a user browsing the Worldwide Web to initiate voice communications with a subscriber by clicking on a voice button which appears on a page being viewed. Voice communication control is handled using computer telephony integration (CTI) hardware enabled to launch calls over the Internet, using a Voice over IP (VoIP) gateway or to initiate a Plain Old Telephone Service (POTS) call through the PSTN using dialed (DMTF) or packet (PRI) trunk technology well known in the art. For each voice communication session, at least two calls are made, a first call back from the service provider to the user browsing the WWW and a second forward from the service provider to a service subscriber designated in a rule base to receive the call. The two calls are then bridged within the CTI hardware. This arrangement is advantageous because it eliminates compatibility issues with PSTN switching equipment, enables rapid service development, and permits unequalled flexibility in call setup and redirection, which enables call control to be exercised by either the user or the service subscriber's agent.

FIG. **1** shows a schematic diagram of a communications network, generally indicated by the reference **10** equipped with apparatus in accordance with the invention. The apparatus in accordance with the invention is principally designed for use by a service provider such as an Internet Service Provider (ISP) **12**. Each ISP **12** has an intranet that is connected to the Public Switched Telephone Network (PSTN) **14** in a manner well known in the art. Each ISP **12** intranet is also connected to the Internet **16** in a manner well known in the art. Internet users such as clients **18** and **20** typically access their ISPs using dial-up connections through the PSTN **14**. Alternatively, clients **18**, **20** may connect to their ISPs using cable modems (not illustrated) or the like. In some installations a user may have only one dial-up connection. For example, client **18** has a single dial-up connection **22** which serves the dual function of data transfer for Internet sessions and voice transfer for telephone conversations. Using telephone **24** client **20**, on the other hand, has a dataline **26** for accessing the Internet and a telephone line **28** for voice communications using telephone **30**.

A service subscriber generally indicated by reference **32** contracts an ISP **12** to maintain one or more pages on a Worldwide Web server which may be accessed on the WWW by clients **18**, **20**. The service subscriber **32** is typically a business having one or more business phones **34** and one or more computer workstations **36** connected directly or indirectly to the PSTN. The service subscriber **32** may have an Intranet (not illustrated). In addition, the telephones **34** and accompanying workstations **36** may be grouped in a single location or dispersed in different premises, different cities or even different countries. Using the apparatus and methods in accordance with the invention, there is no requirement to co-locate equipment or personnel as will be explained below in some detail.

The apparatus in accordance with the invention is typically owned and operated by the ISP **12** although it may also be owned and operated by any other institution including a business having adequate demand for the service it provides. The apparatus in accordance with the invention includes at least a web server **38**, a computer telephony server (CTI Server) **40**, a VoIP gateway **42** and CTI hardware **44**. CTI hardware is well known in the art and available from, for example, Dialogic Corporation of Parsippony, N.J., U.S.A. Alternatively, the CTI hardware **44** may be incorporated in the CTI server **40** also in a manner well known in the art.

FIG. **2** is a block diagram of the hardware and software components required to practise a preferred embodiment of the invention. The components located at the ISP include the web server **38**, CTI hardware **44**, the VoIP gateway **42** and the CTI server **40**. In accordance with a preferred embodiment of the invention, there is also provided an advertisement server **46** for downloading advertisements to voice buttons as will be explained below in more detail. A chat server **48** permits text chats between clients **18**, **20** and service subscribers **32** in a manner well known in the art. A Billing server **50** tracks PSTN usage and long distance charges and prepares call detail records for the ISP which uses the call detail records to prepare bills for its service

subscribers. An operations, administration and maintenance server **52** (OAM server) permits the ISP and service subscribers to maintain rule bases which control and customize services provided using the apparatus in accordance with the invention, as will also be explained below in more detail. An industry standard database server **54** stores call detail records and OAM records to facilitate maintenance and reporting. Typically, billing server **50**, OAM server **52** and DB server **54** are implemented on a single machine although other implementations are possible. Likewise, the CTI hardware **44** and the CTI server **40** are typically integrated into a single machine. The CTI server may also include the chat server **48** which is an "thin server". The Web server **38** will typically also accommodate the advertisement server **46**. Likewise, the VoIP gateway **42** may be integrated into the CTI server **40**.

The servers **38** and **46**–**54** have certain functional requirements. In particular, the voice web server **38** requires no special hardware but must support the ability to download JAVA applets to the clients **18**, **20** workstations and/or the subscriber workstations **34**.

The CTI server **40** must be programmed to handle call setup protocol on receipt of voice communication requests and to pass alerting applets to subscriber agents as will be described below. The CTI server **40** also controls all of the CTI hardware **44** including call setup and tear-down functions. In addition, the CTI server **40** interfaces to the VoIP gateway **42** which may be, for example, a Brooktrout/ANALOGIC® H.323 board commercially available from Brooktrout Technology in Nudham, Mass., U.S.A. It likewise controls all voice connections to the PSTN **14** and controls most of the server logic.

The optional advertisement server **46** is used to feed advertisement script to voice button applets to enhance the visual appeal of the applets and to provide an additional stream of revenue.

The chat server **48** enables text chat sessions between clients **18**, **20** and the service subscriber **32**. The chat server must be programmed to interact with the CTI server **40** to establish a chat session, supervise chat protocol between the clients **18**, **20** and the service subscriber **32** to send chat applets on request. It also handles requests to start text chat sessions and facilitates those sessions between clients **18**, **20** and service subscriber **32** workstations. As noted above, the chat server is a thin server and may be integrated with the CTI server **40**.

The billing server **50** consolidates Telco billing records with voice connections stations established by the apparatus in accordance with the invention and generates call detail records for the ISPs which are used to generate service bills for the subscribers **32**. Suitable billing applications are commercially available and require minimal customization.

The OAM server **52** provides a web interface for ISP and subscriber administration functions. The OAM server provides the tools to create subscribers, subscriber access and to create, store and maintain the rules which customize the functions of the apparatus to serve each particular subscriber's needs. Each time a call is placed, the WEB server **38** retrieves relevant rules from the OAM server **52** to determine how the subscriber leg of the call is to be set up.

The database server **54** ports an industry standard ODBC database for storing, logging and call detail records. The DB server **54** also stores the rule base.

FIG. **2** also shows the software requirements for the client **18**, **20** workstations. The voice button used to initiate a voice communication session with the service subscriber **32** may be a voice button form **56** written in CGI scripts that are embedded in the web page code that transmits appropriate call setup messages to the web server **38**. It may likewise be a voice button applet **58** written as a JAVA applet that is loaded at the same time as the web page is loaded into the client **18**, **20** web browser. The applet transmits the appropriate call setup messages to the web server **38**. A voice button hyperlink (not illustrated) may also be used. A hyperlink with the web page that links with appropriate CGI scripts on the web server **38** may be used for this purpose. When accessed, the CGI script on the server transmits the appropriate call setup messages to the web server **38**. If voice over IP communications are to occur, the client **18**, **20** workstation must include a VoIP client **60** that receives call setup requests from the VoIP gateway **42**. If the web server **38** detects that the client **18**, **20** does not have a VoIP client **60** resident, the web server **38** could be enabled to request permission to download an appropriate VoIP client **60**.

The text chat applet **62** provides a text chat form to enable a text chat session with the service subscriber **32**. A text chat applet **62** is only required if a text chat session is requested. It is assumed that in most instances voice communications will be the preferred medium of information exchange.

FIG. **2** also shows the functional requirements required for the service subscriber **32**. The service subscriber **32** requires an administrative function to perform the administration and maintenance required for the service. A call handling agent for the service subscriber **32** may also be an administrator. Administration is accomplished by logging onto the OAM server **52** which downloads administration forms (not illustrated) required to create and maintain rules respecting the handling of calls initiated by the CT server **40** under control of the web server **38**. As will be explained below, the rules established by administration determine how the service subscriber **32** leg of each voice communication is established. All administration is handled using OAM forms **64**.

The service subscriber agents require alert applets **66** and chat applets **68** on their workstations. The alert applets are used to send call alerts to subscriber stations during call setup, as will be explained below with reference to FIG. **5**, and optionally to permit a subscriber agent to accept or refuse any call requests. The chat applets **68** are used to permit text chat sessions, as described above.

FIGS. **3***a* and **3***b* show an example of a voice button **70**. In actual practice, a voice button **70** may have any appearance and the example shown is exemplary only. If a client **18**, **20** clicks on the "Call Us" button **72** shown in FIG. **3***a*, a voice communication request is initiated which prompts a voice communication request form **76** shown in FIG. **4**. It will be understood by those skilled in the art that an interactive information page may have two or more voice buttons, each dedicated to the same or different functions. Alternatively, the service subscriber **32** may wish to direct specific inquiries to different agents and/or different locations using one voice button. In that case, clicking on the "Call Us" button **72** may initiate a drop-down menu **74** (see FIG. **3***b*) which permits the client **18**, **20** to direct the voice communication request to a specific topic or category. The rule bases created by the service subscriber **32** will determine where calls for each category are directed. When a specific category is selected, the user is then presented with a form to determine the preferred medium for voice communication or there is a lookup of a pre-stored preference file commonly called a "cookie" on the client hard drive. In the example of client **18** presented above, the client **18** must either select VoIP by clicking the radio button beside that

selection, or may specify his telephone number but in that instance he is required to disconnect from his ISP before the telephone call can be received. In the example of client **20**, the client may indicate his telephone number and is permitted to remain connected to his ISP during the telephone conversation with the service subscriber **32**.

FIG. **5** is a schematic representation of one potential implementation of a screen display **78** for the workstation **36** of a call handling agent for the service subscriber **32**. In this example, a plurality of buttons are available on a call agent toolbar **80** to permit the call agent to rapidly handle and manipulate calls from users browsing the WWW. It should be understood that the functions described below are exemplary only. Other functions may be created and not all functions described need be implemented in order to practise the methods with the apparatus in accordance with the invention. When a user such as client **18**, **20** initiates a call request, the web server **38** preferably sends an alert applet **66** (see FIG. **2**) to the agent screen display **78**. The alert applet **66** preferably opens an alert window **82** on the screen display **78**. The contents of the alert window **82** will depend on the type of connection established with the user and the information available from the connection, as well as implementation preferences. In the example illustrated, a user John Doe has requested a voice connection over a telephone line using telephone number 613-123-4567. The user name is identified from the telephone number and displayed to the call handling agent. Also provided is the URL **84** of the page which the user John Doe was viewing at the time he placed the call request. The URL **84** is a hyperlink. If the call agent wishes to view the same page that John Doe is viewing, he may click on the URL which will display that page on his workstation.

As explained above, the toolbar **80** includes a plurality of buttons which enable various call handling functions. For example, the toolbar **80** may include buttons which permit the call handling agent to accept the call displayed in the alert window **82** using an accept call button **86**. The call handling agent may also reject the call by using reject call button **88**. If the call is accepted or rejected, it is handled by the web server **38** in accordance with service subscriber **32** rules as will be explained below in more detail.

If a call is in progress, the call handling agent may transfer the call using a transfer call button **90** to hand the call off to any other appropriate person as will also be described below in more detail. While a call is in progress, the call handling agent may conference others into the conversation using the conference button **92**. After a conference is accomplished, the call agent may release himself from the call using the "Release Me" button **94**. The call handling agent may also release all parties from the call, including the calling party, using a "Release All" button normally used when a call is completed in order to rapidly free all resources. An option may also exist to release the caller using a "Release Caller" button **98** in an instance where, for example, the caller has raised a question which requires that the call handling agent contact another for advice, in which case the agent may release the caller until he has obtained information appropriate to answer the question and return the call after the information is at hand. "Release and Return" buttons such as **101** and **102** may also be implemented. These buttons may be enabled to release the call handling agent for a period of time while, for example, a caller listens to or views prerecorded audio or multi-media information. During the audio or multi-media broadcast, the call handling agent is released to perform other functions. After the broadcast is completed, the call is automatically returned to the call handling agent to answer any further questions the caller may have. This function permits a business to prerecord answers to, for example, most frequently-asked questions or most frequently-sought information. When convenient and appropriate, the call handling agent may quickly enable a broadcast by using the release and return buttons **100**, **102** on the screen display **78**.

FIG. **6** is a schematic diagram showing a simple call sequence in accordance with the invention. For simplicity of illustration, the user is browsing the WWW using client station **18** and the service subscriber **32** has only one workstation **36** and one business phone **34**. The user initiates a call request from an interactive information page maintained for the service subscriber **32**. The call request is forwarded over the Internet to the web server **38**. The call request may include the preferred call completion medium for the client **18** which is automatically attached to the request by the user's web browser if a preference "cookie" is available on a hard drive of the client **18**. If the request does not include the preferred call completion medium for the client **18**, the web server **38** sends a preference form (see FIG. **4**) to the client **18** requesting the information. On receipt of the call request or the preference form, the web server **38** queries the OAM server **52** to obtain subscriber rules to determine how the call request is to be completed to the subscriber **32**. The web server **38** then passes information to the CTI server **40** which operates the CTI hardware **44** to place a first call to the user client **18**. In this example, the user client **18** has requested a VoIP call connection. The CTI server therefore sends a request to the VoIP gateway **42** to make a VoIP call offer to the user client **18**. The call offer is accepted by the user client **18** and the call is answered. The VoIP gateway **42** informs the CTI server **40** that the call has been answered and the CTI server **40** informs the web server **38** that the call connection is complete. The web server **38** forwards call alert information to the workstation **36** of service subscriber **32**. The call alert information "pops" open a call alert window **82** (see FIG. **5**) on the workstation **36**. Web server **38** then passes information to the CTI server **40** to enable a call to the service subscriber **32**. In this example, the service subscriber has specified that calls be completed over the PSTN to the business telephone **34**. The CTI server **40** therefore initiates a PSTN call. The PSTN call is initiated using, for example, a trunk link to the PSTN having a PRI interface, well known in the art. A PRI packet is therefore sent over the trunk to the PSTN which completes the call in a manner well known in the art. When the service subscriber **32** answers the call on business phone **34**, the CTI server **40** informs the web server **38** that the call is answered. The web server **38** then instructs the CTI server **40** to join the calls and conversation between the user client **18** and the service subscriber **32** ensues. In this instance, the first leg of the call is a VoIP call and the second leg of the call is a PSTN voice connection. If this is a first-time user of the service by the user client **18**, the CTI server may be programmed to make a preference file cookie which it sends to the user client **18**. If the cookie is accepted, the cookie is stored on a hard drive of the user client **18** where it may be used to in future determine the voice connection medium preference and other variables related to a connection with the user client **18**. After conversation between the user and the service subscriber is complete, the service subscriber **34** will typically disconnect which causes disconnection of the user **18**. The disconnect is effected by the CTI server **40** which requests the VoIP gateway **42** to disconnect the VoIP session.

FIG. **7** illustrates a more complicated call distribution option enabled by the methods and apparatus in accordance

with the invention. This option is referred to as the use of "simultaneous notification groups" (SNGs). In this example, service subscriber **32** has four call handling agents which use business phones **34A–34D** and workstations **36A–36D**. The call handling agents may be located in the same premises or may be located anywhere that is convenient so long as appropriate facilities are available. For the sake of clarity, the initial call request is not shown in FIG. **7**. A described above, user client **18** initiates a VoIP call request and the web server **38** in cooperation with the CTI server **40** and the VoIP server **42** set up a VoIP call to the user client **18**. The subscriber rules retrieved by the web server **38** (see FIG. **6**) indicate that call attempts are to be made simultaneously to four service subscriber agents defined in an SNG. As described above, a call alert is sent to each of workstations **36A–36D** and PRI packets are sent over the PSTN trunk connection to substantially simultaneously initiate calls to call handling agents A, B, C and D in the SNG. SNGs are particularly useful if it is unpredictable as to which of several agents may be available to take a call, but a rapid response to user calls is desired. In this example, B answers the call first, which is sensed by the CTI server **40**. CTI server **40** informs web server **38** that B has answered the call. Web server **38** instructs the CTI server **40** to drop calls to A, C and D and to join the VoIP call to user client **18** with the PSTN call to call handling agent B of service subscriber **32**. Conversation between user client **18** and call agent B then ensues. If appropriate, a cookie is made and sent from the CTI server **40** to the user client **18**. After the call is completed, call handling agent B disconnects which causes the CTI server **40** to disconnect the VoIP call to user client **18**.

In addition to the SNGs described above, the invention may be used to implement a feature called "agent locator" in which the calls to A, B, C and D are, for example, separate numbers for the same individual. This permits a call handling agent which circulates between premises or the like to be reached regardless of his location. This feature may be further enhanced if a party who answers the call is prompted to accept the call request by dialing a given sequence. For example, "This call is for John Smith, please accept by pressing 1 now.". This minimizes the probability that a call will be accepted by the wrong call handling agent and greatly enhances the flexibility of the service agent locator feature. If the agent locator feature is used, it may be desirable to structure the rule base so that only certain numbers are dialed in a first connection attempt. For example, it may be desirable to dial simultaneously a first and second work number, a home number and a cellular phone number. A pager number is not dialed, however, because it is programmed to answer calls on a first ring and the agent would receive a page even though he is available at one of the other numbers to take the call. The proper structuring of rule bases to properly handle call features will be apparent to those skilled in the art.

FIG. **8** shows another call sequence enabled by the method and apparatus in accordance with the invention. In this example, user client **20** initiates a call request to his telephone **24**. On receipt of the request, the web server **38** requests subscriber rules from OAM server **52**. After the rules are obtained, the web server **38** sends information to CTI server **40** instructing the CTI server **40** to call user client **20** by placing a PSTN call to the telephone **24**. The CTI server **40** informs the web server **38** that the call is answered when the user client **20** answers the telephone **24**. Call alert information is then sent by the web server **38** to the subscriber workstation **36** as described above and information obtained from the subscriber rules is sent to the CTI server **40** to instruct the CTI server **40** to call the subscriber **32** unless the call alert is rejected at the subscriber workstation **36**. In this example, the subscriber is unavailable and the subscriber call is not answered. After a predetermined time period, the CTI server **40** informs the web server **38** that no answer has been received from the subscriber call. The web server **38** again consults the subscriber rules and instructs the CTI server to hold the user while no answer options are sent to the user client **20**. While the user is on hold, the CTI server **40** may be enabled to broadcast music or information to the subscriber telephone **24** in a manner well known in the art. The "no answer" options downloaded to the user client **20** may, for example, include:

1) Leave a text message;
2) Leave a voice mail message;
3) Call me back at a specified time.

In this example, the user client **20** elects to leave a voice mail message and selects option **2** . The option is sent to the web server **38** which instructs the CTI server **40** to connect the user to the voice mail system of the service subscriber **32**. The CTI server **40** places a call to the voice mail system of the service subscriber **32** and the user leaves the voice mail message in a manner that is well understood. When the message is completed, the user hangs up which sends a disconnect signal to the CTI server **40** causing the CTI server **40** to disconnect from the service subscriber's voice mail.

FIGS. **9***a* and **9***b* show partial call sequences which further illustrate the flexibility and utility of the methods and apparatus in accordance with the invention. In this example, a VoIP call is established with user client **18** and a PSTN call is established with call handling agent B of service subscriber **32**. During the conversation, the user asks questions which B determines are better answered by call handling agent D. The call handling agent B therefore selects the transfer call button **90** (see FIG. **5**) on his screen display **78** which initiates a transfer function form (not illustrated) to enable the call to be transferred. The user B completes the form which is automatically sent to the web server **38**. On receipt of the form, the web server **38** instructs the CTI server **40** to hold the call to user client **18**. The CTI server **40** responds that the user client **18** is on hold. The web server **38** then instructs the CTI server **40** to disconnect B. The CTI server **40** disconnects B and confirms the disconnection to the web server **38**. Web server **38** then instructs the CTI server **40** to call agent D. The CTI server **40** calls agent D. When call handling agent D answers, the CTI server **40** informs the web server **38** that the call has been answered and the web server **38** instructs the CTI server **40** to join the calls. Conversation between the user and call handling agent D then ensues. Call handling agent D subsequently, for example, learns that information is requested which is best supplied by call handling agent A. D therefore sends a request to the web server **38** using the conference call button **92** (see FIG. **5**) to conference in A. On receipt of the conference call request form (not illustrated) the web server **38** instructs the CTI server **40** to call A. When A answers, the CTI server **40** confirms to the web server **38** that A has answered and the web server **38** responds by instructing the CTI server **40** to join A to D and the user client **18**. A three-way conversation between the user client **18** and call agents D and A ensues. D thereafter determines that he has no further contribution to make to the conversation so using the Release Me call button **94** on his agent screen display **78** (see FIG. **5**) he sends a release request to the web server **38** which instructs the CTI server **40** to release D. CTI server

34 releases caller D and confirms the release to the web server 38. Thereafter conversation with A ensues. During that conversation, A determines that information from E and F are required to supply all the requests of user client 18. E and F are, for example, experts available at a related company. Call handling agent A therefore uses his conference call button 92 to send a conference request for callers E and F. On receipt of the request, the web server 38 instructs the CTI server 40 to call E. The CTI server 40 places a PSTN call to E and E answers the call. The CTI server 40 informs the web server 38 that the call has been answered and the web server 38 instructs the CTI server 40 to join E to the conversation between the user and call handling agent A. The web server 38 then instructs the CTI server 40 to call F. It will be understood by those skilled in the art that E and F could have been called simultaneously as well. In this example, the CTI server 40 calls F but determines that F's phone is busy. The CTI server 40 informs the web server 38 that F is busy and the web server 38 instructs the CTI server 40 to play an announcement to the user, call agent D and E. The CTI server 40 then uses a prerecorded message or a voice synthesizer to announce that the fourth party is busy and cannot be joined to the call. Call disconnection proceeds as described above. This sequence is illustrative only of the flexibility in call handling enabled by the apparatus and methods in accordance with the invention.

FIG. 10 illustrates yet another feature enabled by the methods and apparatus in accordance with the invention called "automated page return". Only a portion of the call sequence is illustrated in FIG. 10. The call sequence begins after the web server 38 has instructed the CTI server 40 to call the subscriber 32. In this example, subscriber 32 is not available and does not answer the telephone 34. After a predetermined time, the CTI server 40 informs web server 38 that the subscriber has not answered. In response, the web server 38 instructs the CTI server 40 to hold the user (not illustrated) and sends an options list to the user client 18. Among the options is an option to page the subscriber. The user client 18 selects the page subscriber option and returns the option request to the web server 38. On receipt of the option request, the web server 38 again instructs the CTI server 40 to hold the user and to generate a call identification number which will be used to identify the call connection to the user client 18. The identification number is recorded in a table together with a port number, for example, to identify the call connection in a manner well known in the art. The web server 38 then instructs the CTI server 40 to play an on-hold broadcast to the user client 18. The on-hold broadcast may be recorded music, audio advertising, or multimedia information or advertising, for example. The CTI server 40 begins the broadcast over the user call connection. The web server 38 then supplies a pager number from the rule base and instructs the CTI server to page the subscriber. The CTI server 40 dials the pager number using the PSTN and when the paging service answers the CTI server 40 sends the call identification number, which is displayed on the subscriber's pager. The subscriber responds by calling a predetermined number that connects the subscriber to a port of the CTI hardware 44. On connection, the subscriber enters the call ID, for instance the number "1". On receipt of the call ID, the CTI server 40 informs the web server 38 that the subscriber has answered. The web server 38 instructs the CTI server 40 to stop the broadcast to the user client 18 and to join the calls using the call ID to locate the user client 18. The CTI server 40 joins the paged subscriber with the user client 18 enabling conversation to ensue. Call disconnect sequences are as described above. The automated page return feature enables a paged subscriber to be connected directly to a caller using a simple call identification number rather than dialing the user directly as is normal for other paging services. This permits the service subscriber 32 to take advantage of discount call rates available through the service provider. It also enables the return of paged calls to a VoIP connection which is otherwise impossible using prior art methods.

Many other features and functions are also enabled by this invention. For example, a business may wish to be able to distinguish calls from established clients from those of potential clients. This may be accomplished in several ways. For example, two voice buttons may be used. One of the voice buttons may require an access code such as a password, or may be placed on a subordinate web page that requires a password. Calls from each voice button may be directed to a different termination, or they may be directed to different numbers which terminate at the same address but cause different ringing patterns to permit call agents to determine the origin of the call.

The various call features and sequences described above are intended to be exemplary only. It is substantially impossible to provide a comprehensive description of the features enabled by the method and apparatus in accordance with the invention. The explicit examples described are therefore exemplary only and are not to be taken as limiting the scope of this invention.

We claim:

1. A method of providing voice communications between two parties using computer controlled telephony hardware which is outside the Public Switched Telephone Network (PSTN), one of the parties having access to a data network, comprising:

accepting a voice communication request initiated by the party having selected a voice button to initiate the voice communication request;

originating a first voice connection from the computer controlled telephony hardware outside the PSTN under control of a computer telephony server in response to the voice communication request made by the party having access to the data network, the voice communication request received from the data network including information to permit first and second voice connections to be established;

originating a second voice connection from the computer controlled telephony hardware outside the PSTN under the control of the computer telephony server in response to the voice communication request received from the data network using the information respecting the second voice connection; and

bridging together the first and second voice connections using the computer controlled telephony hardware outside the PSTN under the control of the computer telephony server to permit voice communications between the two parties.

2. A method as claimed in claim 1 wherein the first voice connection is a voice over Internet protocol connection and the second voice connection is a voice call placed through the PSTN.

3. A method as claimed in claim 1 wherein each of the first and second voice connections are voice calls placed through the PSTN.

4. A method as claimed in claim 1 wherein each of the first an second voice connections are voice over Internet protocol connections.

5. A method as claimed in claim 1 wherein a plurality of second voice connections are attempted after the first voice

connection is established, and each of the plurality of voice connection attempts are cancelled except a first of the second voice connection attempts to be answered.

6. A method as claimed in claim 1 wherein if the first voice connection is originated but the second voice connection cannot be originated, the first voice connection is held while alternate communications options are downloaded over the data network to the called party of the first voice connection.

7. A method as claimed in claim 6 wherein the communications options may include at least one of an option to leave a voice mail message, an option to leave a text message, and option to request a call back at a later time and an option to page the service subscriber.

8. A method as claimed in claim 7 wherein if the called party of the first voice connection selects the option of leaving a voice mail message, the computer controlled telephony hardware outside the PSTN is operated to establish a second voice connection to a voice mail system and the first and second voice connections are bridged together at the computer controlled telephony hardware.

9. A method of providing voice communication between a service subscriber and a user of a data network accessing an interactive information page available on the network using a data terminal, comprising the steps of:

a) accepting at a voice button server a voice communication request initiated by the user using a voice communication request button associated with the service subscriber and available on the interactive information page;

b) analyzing at the voice button server the voice communication request, or a preference file sent from the user's data terminal, to determine a voice connection medium preferred by the user;

c) consulting a rule base using a code that identifies the subscriber to locate at least one rule that determines how a voice connection is to be established with the subscriber;

d) operating a computer telephony server to control computer telephony hardware outside the PSTN to place a first call to the user by the preferred voice connection medium;

e) operating the computer telephony server to control the computer telephony hardware outside the PSTN to place a second call to the subscriber in accordance with the at least one rule located in the rule base;

f) operating the computer telephony server to control the computer telephony hardware outside the PSTN to bridge the first and second calls;

g) monitoring the data network for an indication that a modification of a configuration of the voice communication is desired by either the user or the subscriber and changing the configuration of the voice communication if an indication for change is received; and

h) monitoring the first and second voice calls and disconnecting the other of the first and second calls when either of the first or second calls is disconnected.

10. A method as claimed in claim 9 wherein the data terminal is a multimedia computer, the voice connection medium preferred by the user is voice over the data network and the first call is a voice over the data network connection to the data terminal.

11. A method as claimed in claim 9 wherein the voice connection medium preferred by the user is a telephone call through the switched telephone network and the first call is routed by the computer telephony hardware outside the PSTN to the switched telephone network.

12. A method as claimed in claim 9 wherein the rule b indicates that the call is to be connected to any one of a plurality of subscriber terminations and a plurality of second call attempts are initiated to the plurality of subscriber terminations, the first call being bridged to a first subscriber termination to answer one of the second call attempts, the remainder of the second call attempts being cancelled by the computer telephony hardware outside the PSTN when the first of the second calls is answered.

13. A method as claimed in claim 12 wherein the plurality of subscriber terminations are telephone numbers or IP addresses for different agents designated t accept the call.

14. A method as claimed in claim 12 wherein each of the subscriber terminations are different telephone numbers or IP addresses for the same agent designated to accept the call.

15. A method as claimed in claim 14 wherein a person answering a call to any of the subscriber terminations is prompted to enter a code to accept the call.

16. A method as claimed in claim 14 wherein the different telephone numbers comprise a work number, a home number, a cellular number, and a pager number; and the different IP addresses are for a work terminal and a home terminal.

17. A method as claimed in claim 16 wherein the at least one rule dictates that the pager number is to be dialed only if the work number, the home number and the cellular number fail to answer after a predefined number of rings.

18. A method as claimed in claim 9 wherein the voice communication request button includes a menu of voice communication request targets and the rule base selects one or more voice connections for the second call based on the voice communication request target selected by the user.

19. A method as claimed in claim 9 wherein the information page or a subordinate page associated with the information page includes a second voice communication request button that may only be accessed or operated by users in possession of a predefined access code.

20. A method as claimed in claim 19 wherein the rule base is structured to treat calls which originate from the second voice communication request button differently from those that originate from the first voice communication request button.

21. A method as claimed in claim 20 wherein the second calls for call requests that originate from the first voice communication request button are directed to a different termination than the second calls that originate from the second call request button.

22. A method as claimed in claim 20 wherein second calls for call requests that originate from both the first and second voice communication request buttons are routed to the switched telephone network and terminate at the same address on the switched telephone network, but different numbers are used for each of the first and second voice communication request buttons and the switched telephone network is configured to apply a different ringing signal on the subscriber line for each number for the termination, the different ringing signal alerting the subscriber as to which voice communication request voice button was selected.

23. A method as claimed in claim 9 wherein if the second call is not answered after a predefined number of rings, the user is presented with a menu of selections to enable alternate means for communication.

24. A method as claimed in claim 23 wherein the menu includes selections for creating a voice message sending a typed message or entering a request for a call back at a designated time.

25. A method as claimed in claim 24 wherein if the user requests the voice message option, the user is automatically

connected to the subscriber's voice mailbox by initiating a second call to the subscriber's voice mailbox and bridging the second call to the first call.

26. A method as claimed in claim **22** wherein the menu includes a selection for paging the service subscriber.

27. A method as claimed in claim **26** wherein when a request for paging the service subscriber is received, the voice connection of the user is held while a call to the service subscriber's paging service is made and a call identification number is sent to the paged subscriber.

28. A method as claimed in claim **27** wherein the call identification number is used to connect the service subscriber to the user when the paged service subscriber calls the computer telephony hardware and enters the call identification number.

29. A system for providing voice communication between a service subscriber and a user of a data network accessing an interactive information page available on the network using a data terminal, comprising in combination:

at least one computing machine programmed with the functionality of:

a) a voice button server for accepting voice communication requests either directly or indirectly form a user accessing an interactive information page on a data network and selecting a voice request button;

b) an operations, management and maintenance server to permit the establishment and maintenance of rule bases for governing how the voice communication requests are completed to the service subscriber; and

c) a computer telephony server for controlling computer telephony hardware outside the PSTN;

a voice over data network gateway for converting voice data packets to voice telephony format; and

computer telephony hardware outside the PSTN which may be controlled by the computer telephony server to initiate voice connections on a switched telephone network or a voice gateway to a data network, to bridge calls and to respond to a set of predefined voice connection control commands that may originate from either the service subscriber or the user.

30. A system as claimed in claim **29** wherein the computer telephony hardware outside the PSTN comprises a controller and a switch fabric.

* * * * *

EXHIBIT J

US006798786B1

(12) **United States Patent**
Lo et al.

(10) **Patent No.: US 6,798,786 B1**
(45) **Date of Patent: Sep. 28, 2004**

(54) **MANAGING CALLS OVER A DATA NETWORK**

(75) Inventors: **Wing F. Lo**, Plano, TX (US); **Xuewen Li**, Cary, NC (US); **Alireza Abaye**, Plano, TX (US)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/370,984**

(22) Filed: **Aug. 20, 1999**

Related U.S. Application Data

(60) Provisional application No. 60/137,877, filed on Jun. 7, 1999.

(51) **Int. Cl.**[7] **H04Q 7/00**; H04J 3/16

(52) **U.S. Cl.** **370/468**; 370/329; 370/392; 370/352; 709/226

(58) **Field of Search** 370/352–360, 370/229–236.1, 329, 389, 392, 341, 468; 379/88.1, 900; 709/224, 226, 238

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,115,429	A	*	5/1992	Hluchyj et al.	370/231
5,526,353	A	*	6/1996	Henley et al.	370/392
5,633,861	A	*	5/1997	Hanson et al.	370/232
5,828,666	A		10/1998	Focsaneanu et al.	
5,870,561	A	*	2/1999	Jarvis et al.	709/238
5,883,891	A	*	3/1999	Williams et al.	370/356
5,889,774	A	*	3/1999	Mirashrafi et al.	370/352
5,907,610	A	*	5/1999	Onweller	370/352
5,953,322	A	*	9/1999	Kimball	370/328
6,046,678	A	*	4/2000	Wilk	340/686.1
6,069,890	A	*	5/2000	White et al.	370/352
6,222,829	B1	*	4/2001	Karlsson et al.	370/329
6,356,545	B1	*	3/2002	Vargo et al.	370/355
6,373,835	B1	*	4/2002	Ng et al.	370/352
6,426,942	B1	*	7/2002	Sienel et al.	370/235
6,426,955	B1	*	7/2002	Dalton et al.	370/401
6,430,289	B1	*	8/2002	Liffick	379/900
6,463,044	B1	*	10/2002	Seo	370/329
6,477,143	B1	*	11/2002	Ginossar	370/230
6,501,736	B1	*	12/2002	Smolik et al.	370/252

FOREIGN PATENT DOCUMENTS

WO WO 99/23799 * 5/1999 .... H04L/29/06

OTHER PUBLICATIONS

International Business Machines Corp., *ATM Overview*, pp. 1–4 (1995).

Y. Bernet, et al., *A Framework for End–to–End QOS Combining RSVP/Intserv and Differentiated Services*, Internet Engineering Task Force Internet Draft, pp. 1–15, (Mar. 1998).

R. Braden, et al., *Resource Reservation Protocol (RSVP)*, Version 1 Functional Specification, pp. 1–62, Request for Comments 2205 (Sep. 1997).

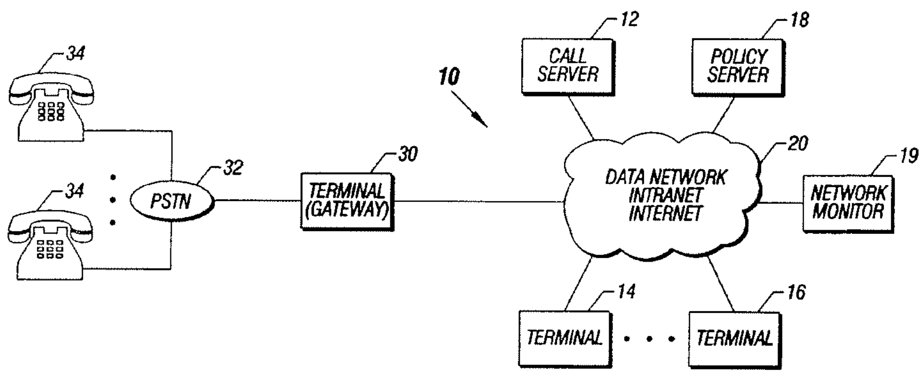
(List continued on next page.)

Primary Examiner—Afsar Qureshi
(74) *Attorney, Agent, or Firm*—Trop, Pruner & Hu, P.C.

(57) **ABSTRACT**

A method and system of managing calls over a data network includes determining an available bandwidth of the data network. After a call request is received for establishing a call between at least two network terminals, one or more of a plurality of resource elements are selected in response to the call request based on the bandwidth of the data network. The resource elements, which can include codecs (coders/decoders), packet sizes (for carrying audio data), and others, are used in the requested call between the at least two network terminals. Further, a plurality of communities may be defined each including one or more terminals. One or more usage threshold values may be assigned to a link or links between communities, and a call request is processed based on the one or more usage threshold values. The processing includes at least one of determining whether to admit the call request and selecting resource elements to be used during a call between terminals over the link.

32 Claims, 11 Drawing Sheets



OTHER PUBLICATIONS

Databeam Corp., *A Primer on the H–323 Series Standard*, Version 2.0, pp. 1–21, printed from web site http://www-.databeam.com, dated at least as early as Jul. 2, 1999.

Trillium Digital Systems, Inc., *Trillium IP Quality of Service White Paper*, pp. 1–37 (Apr. 17, 1998).

J. Toga, et al., *Demystifying Multimedia Conference Over the Internet Using the H.323 Set of Standards*, pp. 1–11, Intel Technology Journal (Second Quarter, 1998).

The Applied Technologies Group, *Wireless Enterprise Networking*, The Technology Guide Series, 1–40 (1998).

* cited by examiner

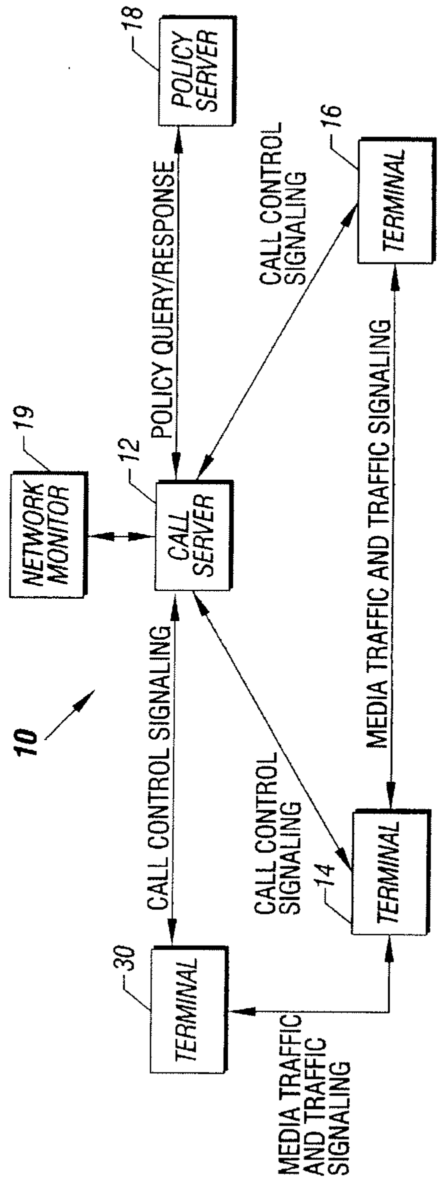

10
NETWORK MONITOR
19
CALL SERVER
12
POLICY SERVER
18
POLICY QUERY/RESPONSE
CALL CONTROL SIGNALING
TERMINAL
16
MEDIA TRAFFIC AND TRAFFIC SIGNALING
CALL CONTROL SIGNALING
CALL CONTROL SIGNALING
TERMINAL
14
TERMINAL
30
MEDIA TRAFFIC AND TRAFFIC SIGNALING

FIG. 1A

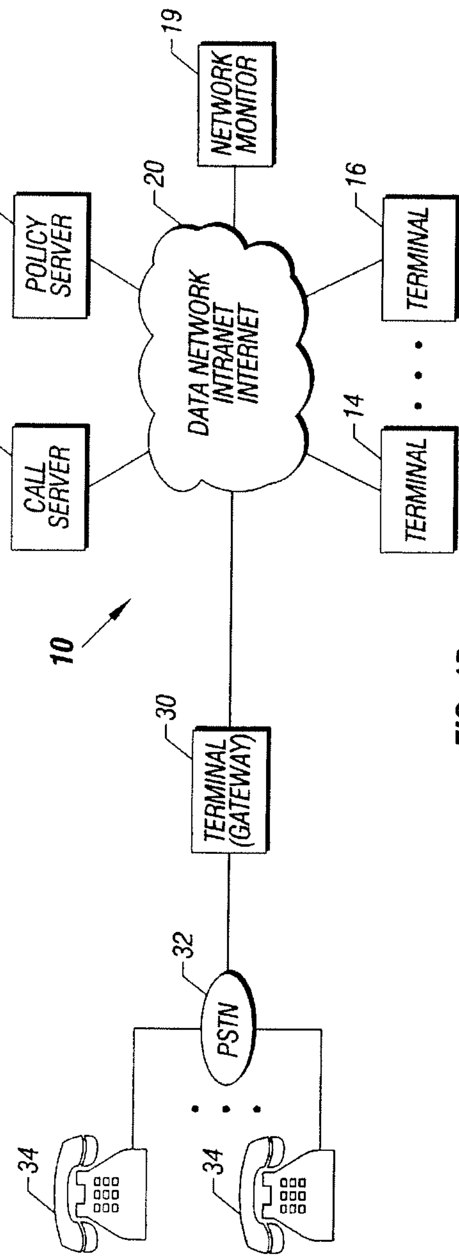

10
CALL SERVER
12
POLICY SERVER
18
DATA NETWORK INTRANET INTERNET
20
NETWORK MONITOR
19
TERMINAL
16
TERMINAL
14
TERMINAL (GATEWAY)
30
PSTN
32
34
34

FIG. 1B

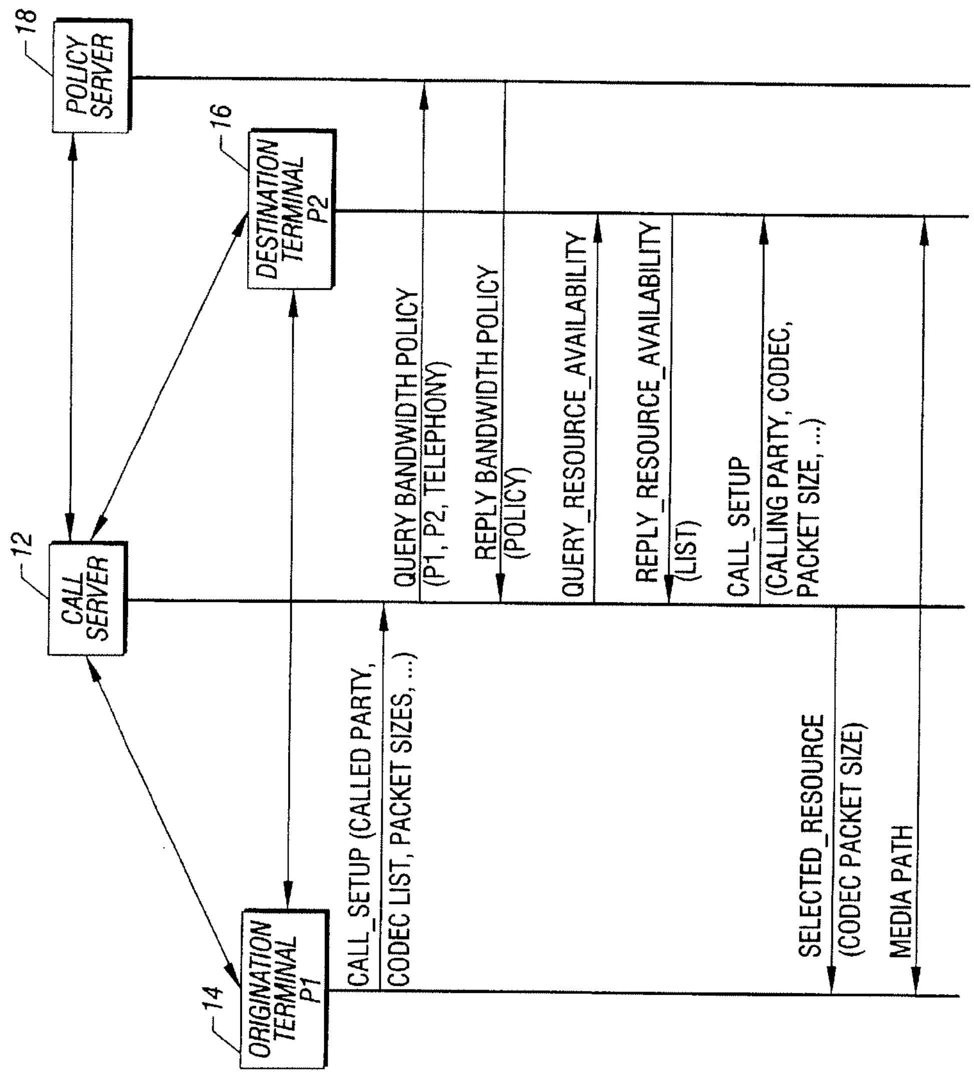

18
POLICY SERVER
16
DESTINATION TERMINAL P2
12
CALL SERVER
14
ORIGINATION TERMINAL P1
CALL_SETUP (CALLED PARTY, CODEC LIST, PACKET SIZES, ...)
QUERY BANDWIDTH POLICY (P1, P2, TELEPHONY)
REPLY BANDWIDTH POLICY (POLICY)
QUERY_RESOURCE_AVAILABILITY
REPLY_RESOURCE_AVAILABILITY (LIST)
CALL_SETUP (CALLING PARTY, CODEC, PACKET SIZE, ...)
SELECTED_RESOURCE (CODEC PACKET SIZE)
MEDIA PATH

FIG. 2

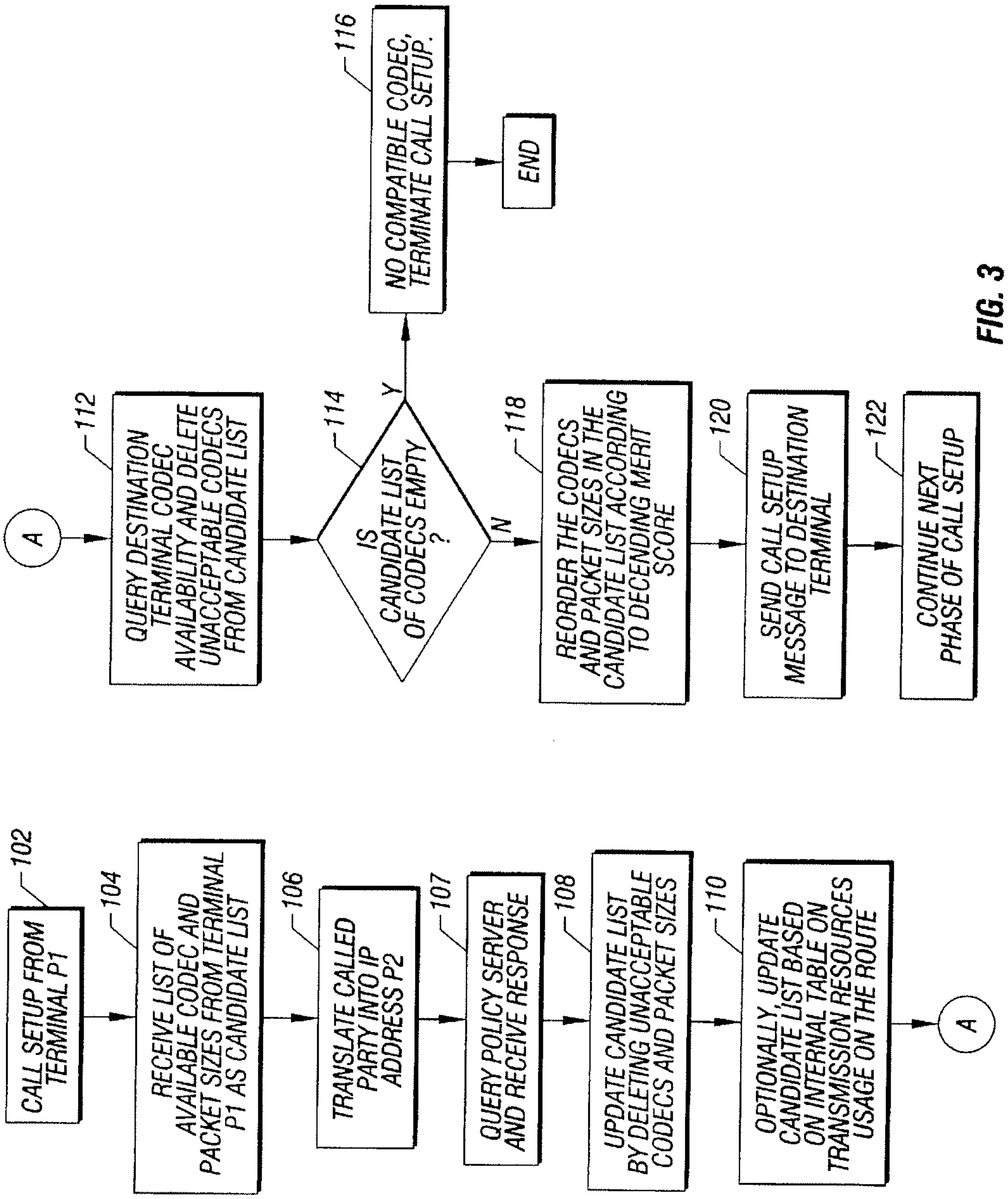

CALL SETUP FROM TERMINAL P1
102
RECEIVE LIST OF AVAILABLE CODEC AND PACKET SIZES FROM TERMINAL P1 AS CANDIDATE LIST
104
TRANSLATE CALLED PARTY INTO IP ADDRESS P2
106
QUERY POLICY SERVER AND RECEIVE RESPONSE
107
UPDATE CANDIDATE LIST BY DELETING UNACCEPTABLE CODECS AND PACKET SIZES
108
OPTIONALLY, UPDATE CANDIDATE LIST BASED ON INTERNAL TABLE ON TRANSMISSION RESOURCES USAGE ON THE ROUTE
110
A
A
QUERY DESTINATION TERMINAL CODEC AVAILABILITY AND DELETE UNACCEPTABLE CODECS FROM CANDIDATE LIST
112
IS CANDIDATE LIST OF CODECS EMPTY ?
114
Y
N
NO COMPATIBLE CODEC, TERMINATE CALL SETUP.
116
END
REORDER THE CODECS AND PACKET SIZES IN THE CANDIDATE LIST ACCORDING TO DECENDING MERIT SCORE
118
SEND CALL SETUP MESSAGE TO DESTINATION TERMINAL
120
CONTINUE NEXT PHASE OF CALL SETUP
122

FIG. 3

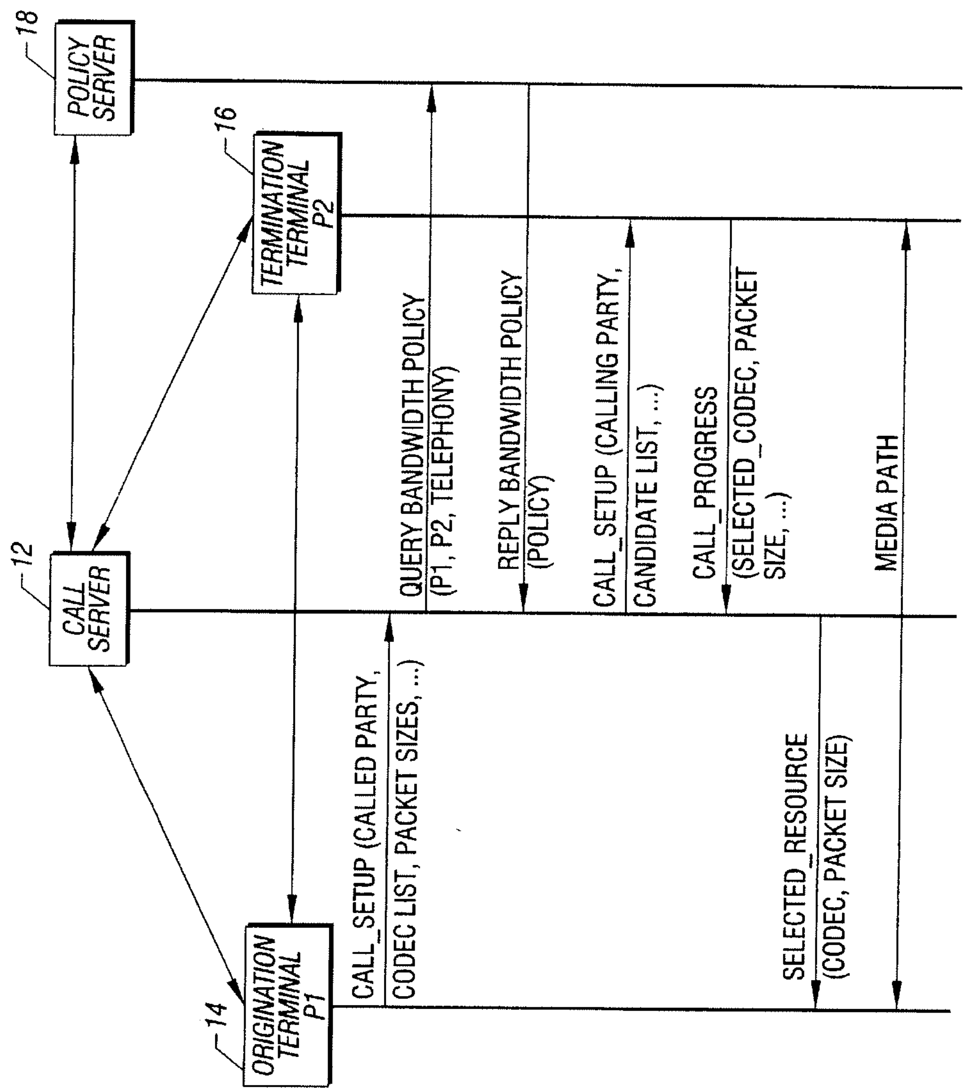

POLICY SERVER
18
TERMINATION TERMINAL P2
16
CALL SERVER
12
ORIGINATION TERMINAL P1
14
QUERY BANDWIDTH POLICY (P1, P2, TELEPHONY)
REPLY BANDWIDTH POLICY (POLICY)
CALL_SETUP (CALLING PARTY, CANDIDATE LIST, ...)
CALL_PROGRESS (SELECTED_CODEC, PACKET SIZE, ...)
MEDIA PATH
CALL_SETUP (CALLED PARTY, CODEC LIST, PACKET SIZES, ...)
SELECTED_RESOURCE (CODEC, PACKET SIZE)

FIG. 4

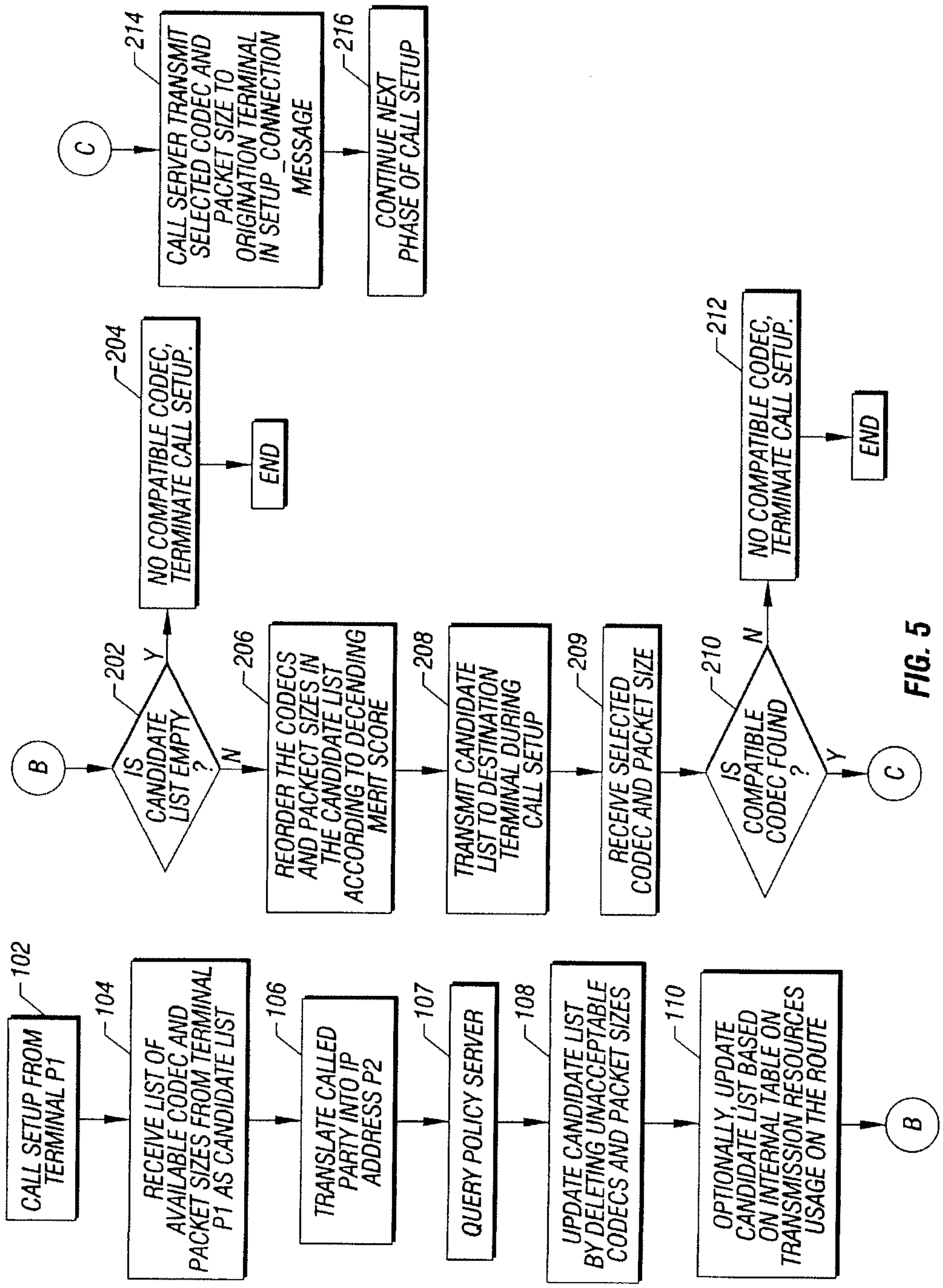

CALL SETUP FROM TERMINAL P1
102
RECEIVE LIST OF AVAILABLE CODEC AND PACKET SIZES FROM TERMINAL P1 AS CANDIDATE LIST
104
TRANSLATE CALLED PARTY INTO IP ADDRESS P2
106
QUERY POLICY SERVER
107
UPDATE CANDIDATE LIST BY DELETING UNACCEPTABLE CODECS AND PACKET SIZES
108
OPTIONALLY, UPDATE CANDIDATE LIST BASED ON INTERNAL TABLE ON TRANSMISSION RESOURCES USAGE ON THE ROUTE
110
B
B
IS CANDIDATE LIST EMPTY ?
202
Y
N
NO COMPATIBLE CODEC, TERMINATE CALL SETUP.
204
END
REORDER THE CODECS AND PACKECT SIZES IN THE CANDIDATE LIST ACCORDING TO DECENDING MERIT SCORE
206
TRANSMIT CANDIDATE LIST TO DESTINATION TERMINAL DURING CALL SETUP
208
RECEIVE SELECTED CODEC AND PACKET SIZE
209
IS COMPATIBLE CODEC FOUND ?
210
N
Y
NO COMPATIBLE CODEC, TERMINATE CALL SETUP.
212
END
C
C
CALL SERVER TRANSMIT SELECTED CODEC AND PACKET SIZE TO ORIGINATION TERMINAL IN SETUP_CONNECTION MESSAGE
214
CONTINUE NEXT PHASE OF CALL SETUP
216

FIG. 5

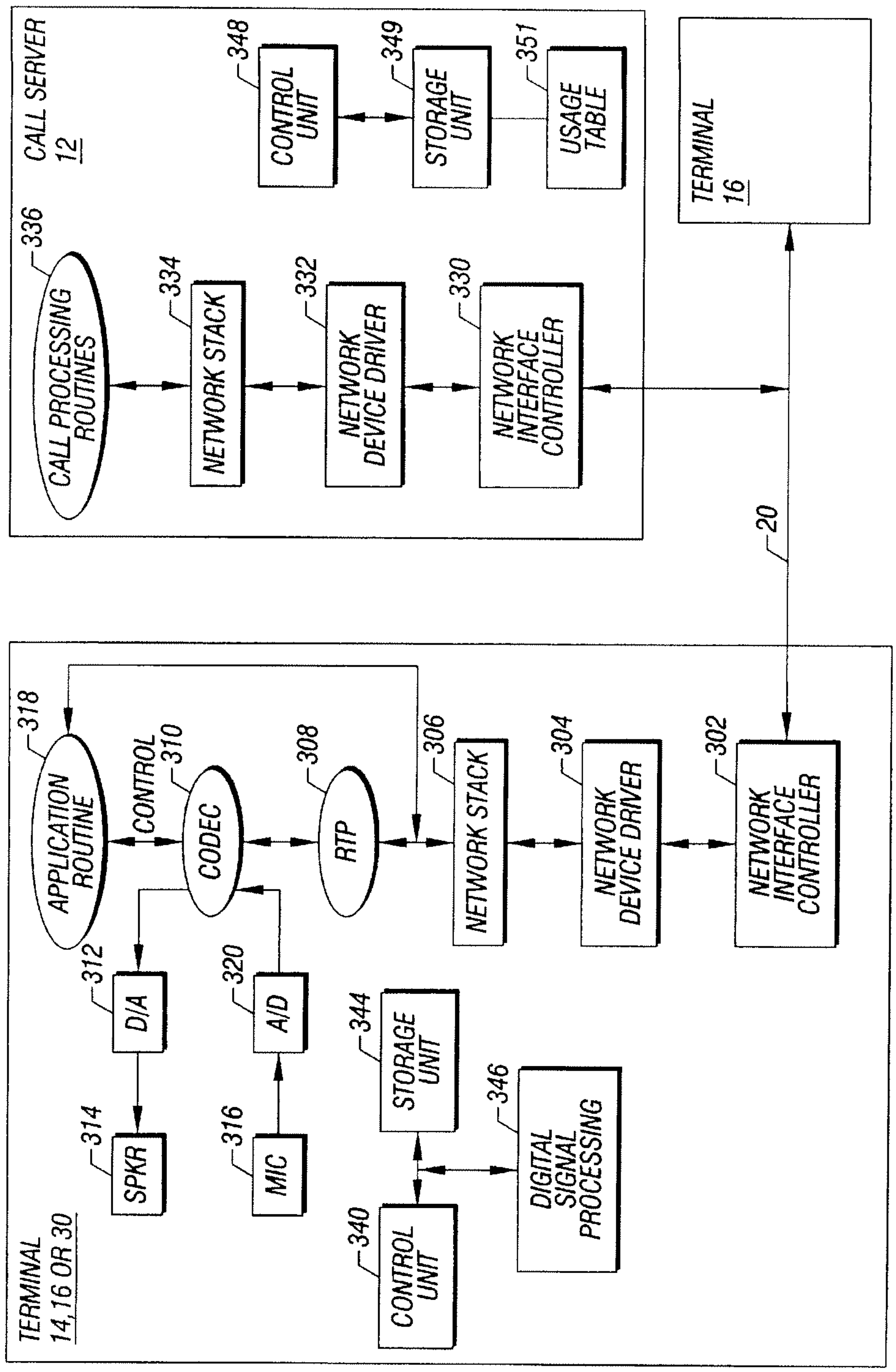
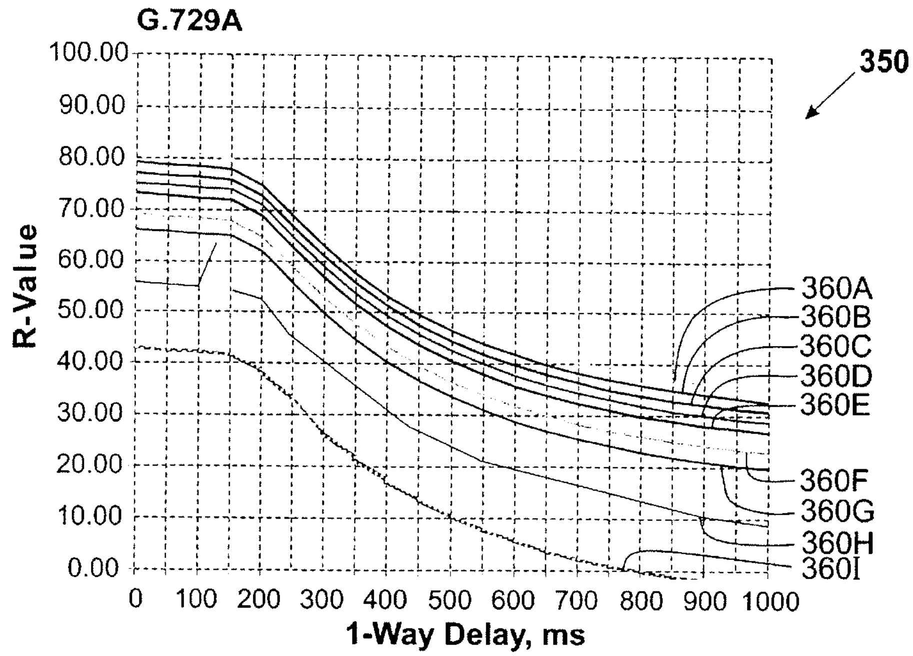
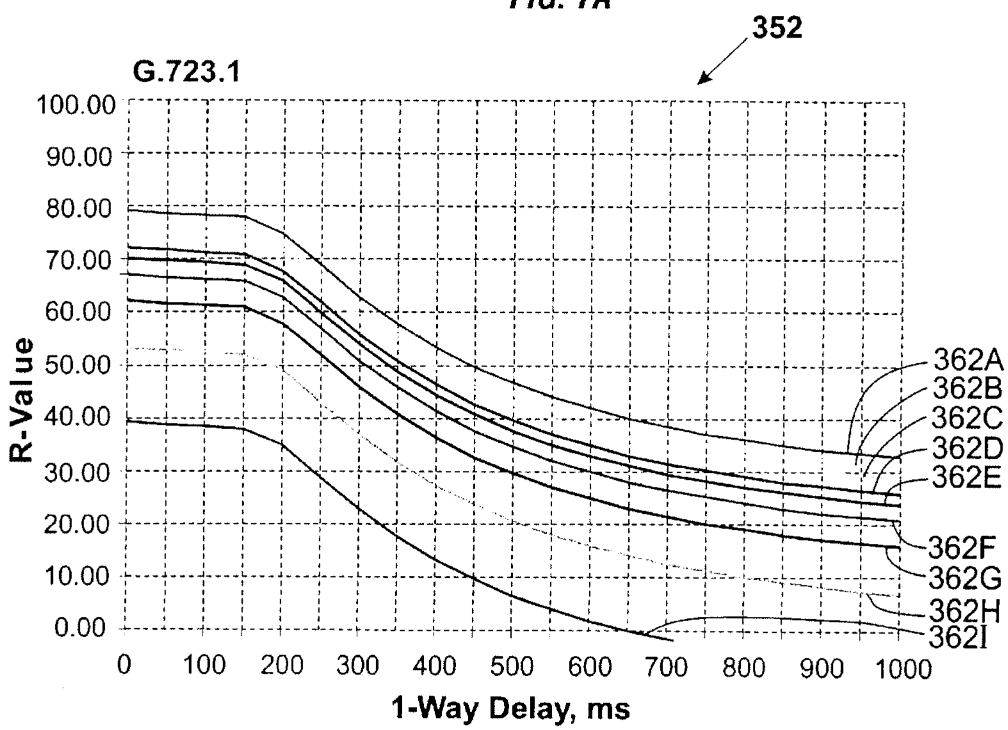

CALL SERVER
12
CALL PROCESSING ROUTINES
336
NETWORK STACK
334
NETWORK DEVICE DRIVER
332
NETWORK INTERFACE CONTROLLER
330
CONTROL UNIT
348
STORAGE UNIT
349
USAGE TABLE
351
TERMINAL
16
20
TERMINAL
14,16 OR 30
APPLICATION ROUTINE
318
CONTROL
CODEC
310
RTP
308
NETWORK STACK
306
NETWORK DEVICE DRIVER
304
NETWORK INTERFACE CONTROLLER
302
D/A
312
SPKR
314
A/D
320
MIC
316
STORAGE UNIT
344
DIGITAL SIGNAL PROCESSING
346
CONTROL UNIT
340

FIG. 6



G.729A
350
R-Value
100.00
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
0
100
200
300
400
500
600
700
800
900
1000
1-Way Delay, ms
360A
360B
360C
360D
360E
360F
360G
360H
360I

FIG. 7A



352
G.723.1
R-Value
100.00
90.00
80.00
70.00
60.00
50.00
40.00
30.00
20.00
10.00
0.00
0
100
200
300
400
500
600
700
800
900
1000
1-Way Delay, ms
362A
362B
362C
362D
362E
362F
362G
362H
362I

FIG. 7B

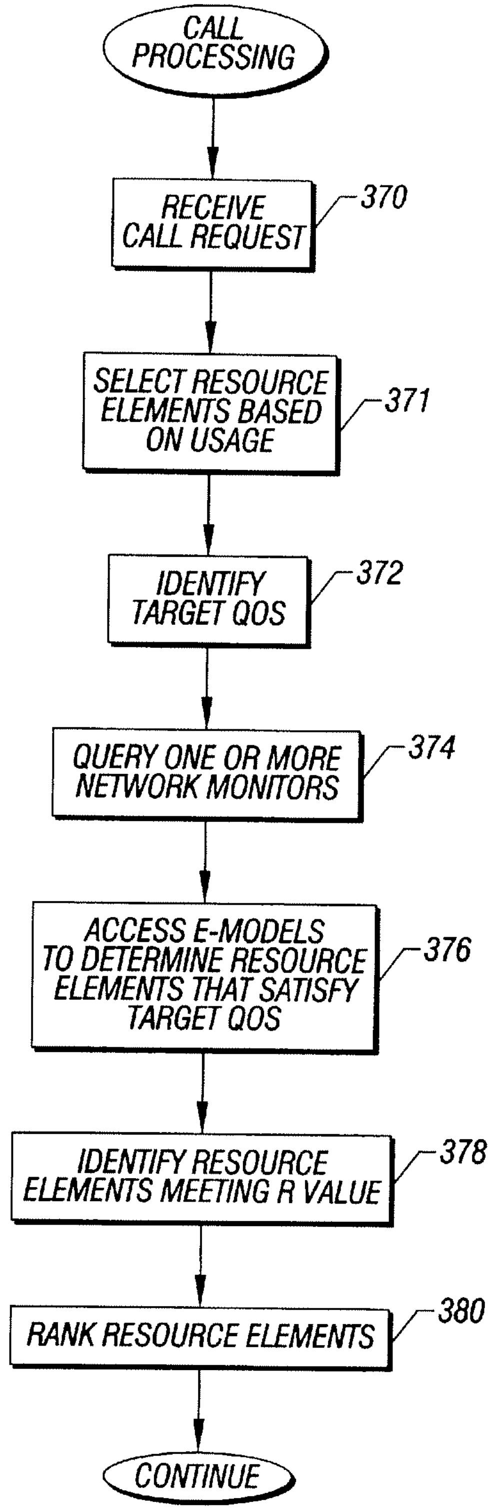

CALL PROCESSING
RECEIVE CALL REQUEST
370
SELECT RESOURCE ELEMENTS BASED ON USAGE
371
IDENTIFY TARGET QOS
372
QUERY ONE OR MORE NETWORK MONITORS
374
ACCESS E-MODELS TO DETERMINE RESOURCE ELEMENTS THAT SATISFY TARGET QOS
376
IDENTIFY RESOURCE ELEMENTS MEETING R VALUE
378
RANK RESOURCE ELEMENTS
380
CONTINUE

FIG. 8

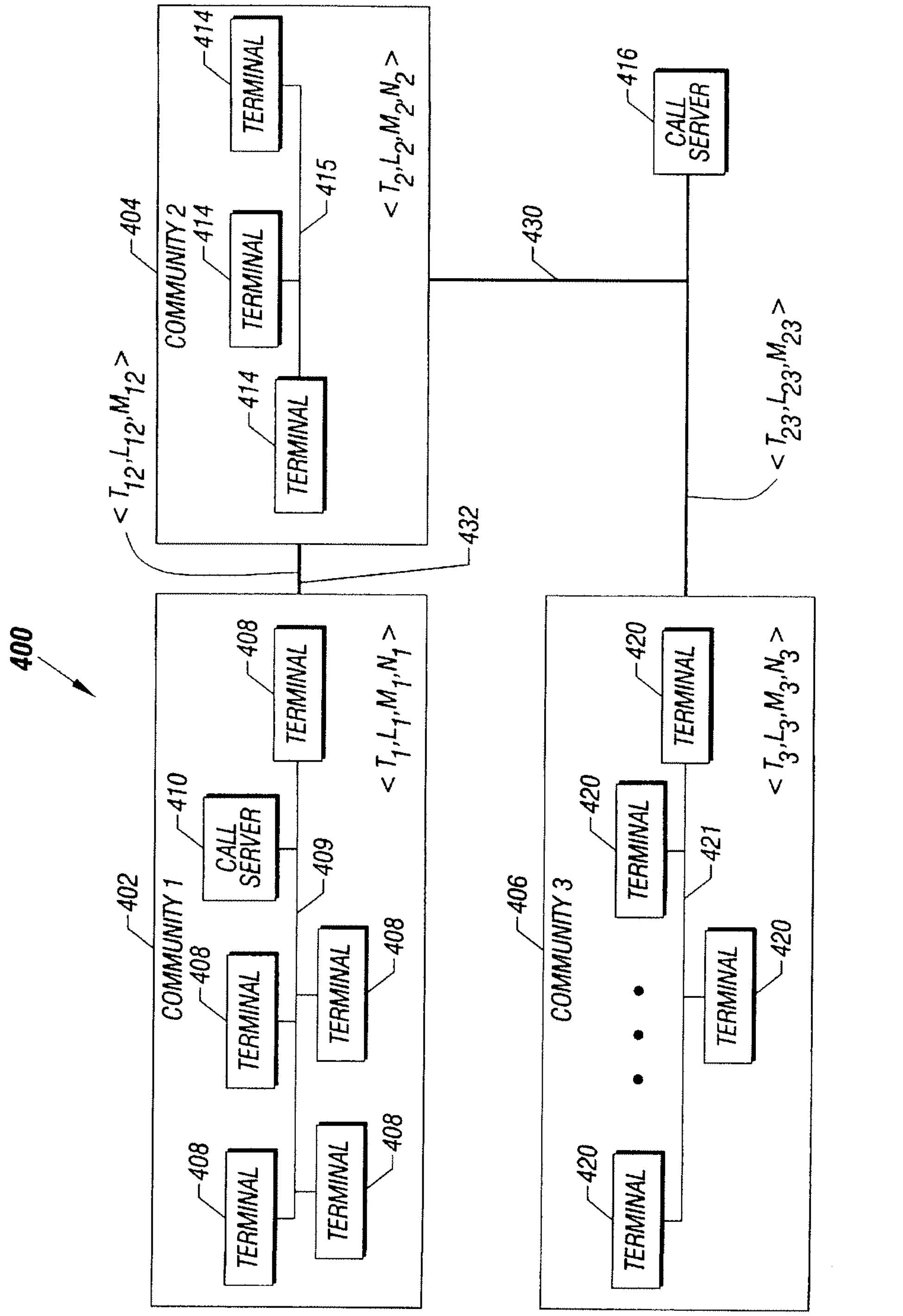
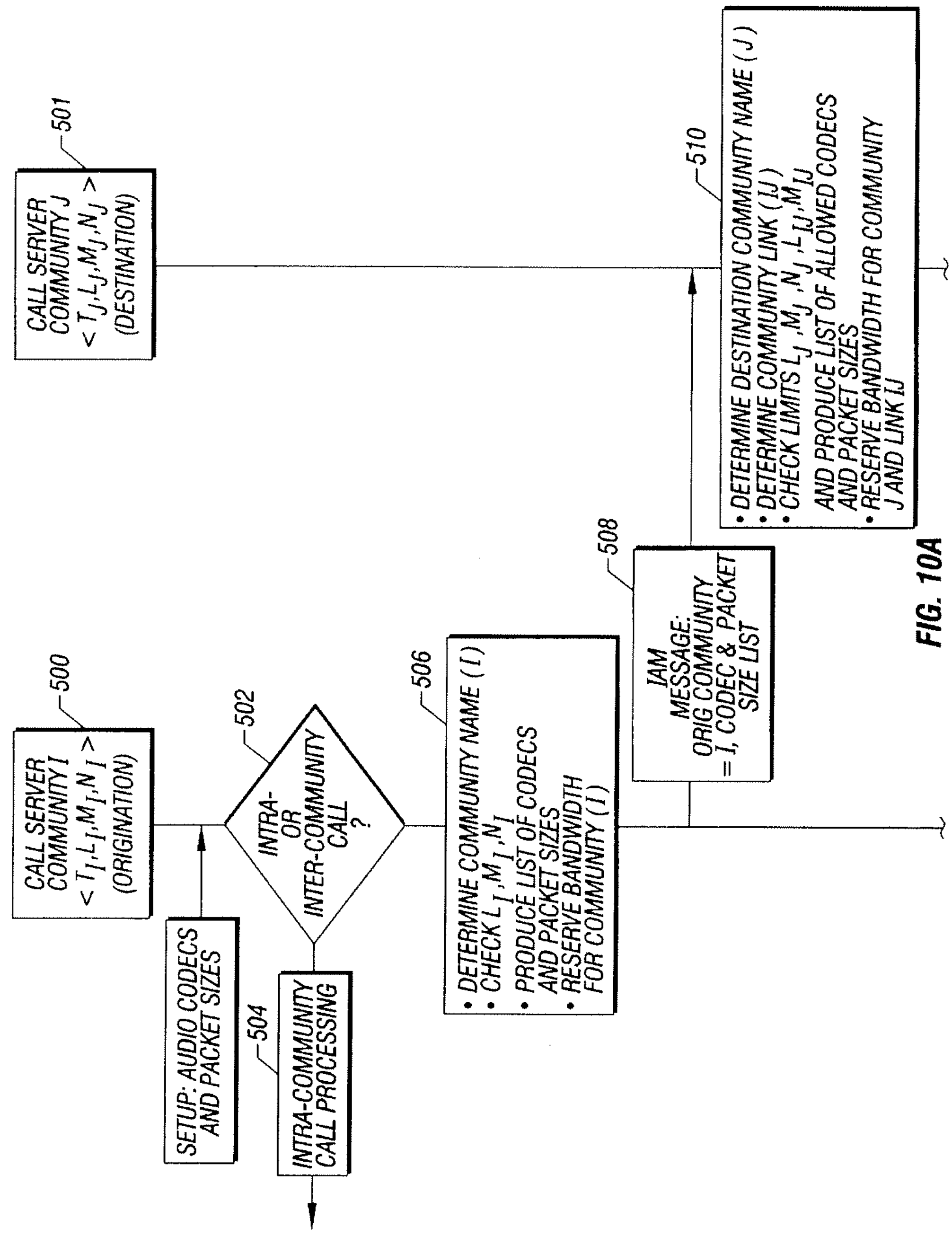

400
COMMUNITY 1
402
TERMINAL
408
CALL SERVER
410
409
$< T_1, L_1, M_1, N_1 >$
COMMUNITY 2
404
414
415
$< T_2, L_2, M_2, N_2 >$
$< T_{12}, L_{12}, M_{12} >$
432
430
416
$< T_{23}, L_{23}, M_{23} >$
COMMUNITY 3
406
420
421
$< T_3, L_3, M_3, N_3 >$

FIG. 9



CALL SERVER COMMUNITY I < T_I, L_I, M_I, N_I > (ORIGINATION)
500
SETUP: AUDIO CODECS AND PACKET SIZES
INTRA- OR INTER-COMMUNITY CALL ?
502
INTRA-COMMUNITY CALL PROCESSING
504
• DETERMINE COMMUNITY NAME (I)
• CHECK L_I, M_I, N_I
• PRODUCE LIST OF CODECS AND PACKET SIZES
• RESERVE BANDWIDTH FOR COMMUNITY (I)
506
IAM MESSAGE: ORIG COMMUNITY = I, CODEC & PACKET SIZE LIST
508
CALL SERVER COMMUNITY J < T_J, L_J, M_J, N_J > (DESTINATION)
501
• DETERMINE DESTINATION COMMUNITY NAME (J)
• DETERMINE COMMUNITY LINK (IJ)
• CHECK LIMITS $L_J, M_J, N_J, L_{IJ}, M_{IJ}$ AND PRODUCE LIST OF ALLOWED CODECS AND PACKET SIZES
• RESERVE BANDWIDTH FOR COMMUNITY J AND LINK IJ
510

FIG. 10A

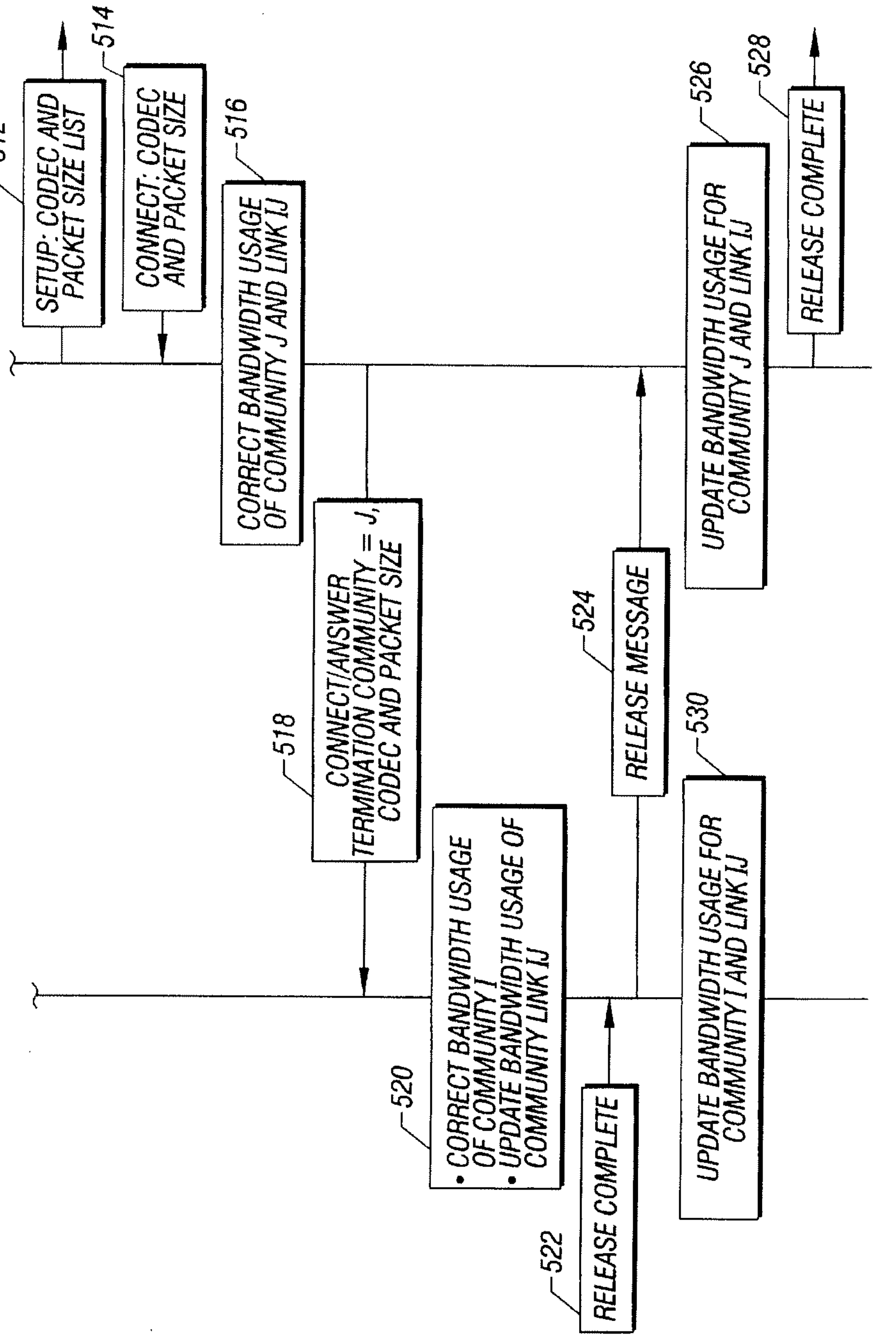

512
SETUP: CODEC AND PACKET SIZE LIST
514
CONNECT: CODEC AND PACKET SIZE
516
CORRECT BANDWIDTH USAGE OF COMMUNITY J AND LINK IJ
518
CONNECT/ANSWER TERMINATION COMMUNITY = J, CODEC AND PACKET SIZE
520
• CORRECT BANDWIDTH USAGE OF COMMUNITY I
• UPDATE BANDWIDTH USAGE OF COMMUNITY LINK IJ
522
RELEASE COMPLETE
524
RELEASE MESSAGE
526
UPDATE BANDWIDTH USAGE FOR COMMUNITY J AND LINK IJ
528
RELEASE COMPLETE
530
UPDATE BANDWIDTH USAGE FOR COMMUNITY I AND LINK IJ

FIG. 10B

MANAGING CALLS OVER A DATA NETWORK

The present application claims priority under 35 U.S.C. §119(e) to U.S. Provisional Patent Application Ser. No. 60/137,877, entitled "Coding Resource Selection for Packet Voice," filed Jun. 7, 1999.

BACKGROUND

The invention relates to managing calls over a data network, such as an Internet Protocol (IP) network.

Packet-based data networks are widely used to link various nodes, such as personal computers, servers, gateways, and so forth. Packet-based data networks include private networks, such as local area networks (LANs) and wide area networks (WANs), and public networks, such as the Internet. The increased availability of such data networks has increased accessibility among nodes, whether the nodes are located in close proximity to each other (such as within an organization) or at far distances from each other. Popular forms of communications across such data networks include electronic mail, file transfer, web browsing, and other exchanges of digital data.

With the increased capacity and reliability of data networks, voice communications over data networks, including private and public networks, have become possible. Voice communications over packet-based data networks are unlike voice communications in a conventional public switched telephone network (PSTN), which provides users with dedicated, end-to-end circuit connections for the duration of each call. Communications over data networks, such as IP (Internet Protocol) networks, are performed using packets that are sent in bursts from a source to one or more destination nodes. Voice data sent over a data network has to share the network bandwidth with conventional non-voice data (e.g., electronic mail, file transfer, web access, and other traffic). One standard that has been implemented for communications of voice as well as other data is the H.323 recommendation from the Telecommunication Sector of the International Telecommunication Union (ITU-T), which describes terminals, equipment and services for multimedia communications over packet-based networks.

In an IP data network, each data packet is routed to a node having destination IP address contained within the header of each packet. Data packets may be routed over separate network paths before arriving at the final destination for reassembly. Transmission speeds of the various packets may vary widely depending on the usage of data networks over which the data packets are transferred. During peak usage of data networks, delays added to the transfer of voice data packets may cause poor performance of voice communications. Voice data packets that are lost or delayed due to inadequate or unavailable capacity of data networks or resources of data networks may result in gaps, silence, and clipping of audio at the receiving end.

A need thus exists for an improved method and system to manage the quality of voice calls or other audio communications over data networks.

SUMMARY

In general, according to one embodiment, a method of managing calls over a data network includes determining usage information of the data network. A call request is received for establishing a call between at least two network terminals. One or more of a plurality of resource elements are selected as candidates for use in the requested call in

response to the call request based on usage information of the data network.

In general, according to another embodiment, a method of managing calls in a telephony system includes defining a plurality of communities each including one or more communication endpoints and assigning one or more usage threshold values to a link between communities. Further, a call request is processed based on the one or more usage threshold values. The processing includes determining whether to admit the call request over the link.

Some embodiments of the invention may provide one or more of the following advantages. Resource elements can be selected to optimize quality of service while at the same time taking into account the usage of the data network as well as usage of other transmission or communications resources. Proper selection of resource elements as well as call admission control reduces the likelihood of overburdening links between terminals. As a result, the likelihood of delays in the communication of audio data that may lead to various audio distortions is also reduced. By efficiently using packet-based data networks for telephony and other forms of audio communications, sharing of such data networks for carrying audio data (which are relatively time sensitive) and traditional forms of digital data (such as electronic mail traffic, file transfer traffic, and other traffic) can be made more effective.

Other features and advantages will become apparent from the following description and from the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

FIGS. **1**A–**1**B are block diagrams of an embodiment of a telephony communications system in which voice or other audio data may be communicated over packet-based data networks.

FIG. **2** illustrates the flow for processing a call request between an origination terminal and a destination terminal in accordance with one embodiment.

FIG. **3** is a flow diagram of tasks performed by a call server in response to a call request in accordance with one embodiment.

FIG. **4** illustrates the flow for processing a call request between an origination terminal and a destination terminal in accordance with an alternative embodiment.

FIG. **5** is a flow diagram of tasks performed by a call server in response to a call request in accordance with the alternative embodiment.

FIG. **6** illustrates components in a terminal and call server of FIGS. **1**A–**1**B.

FIGS. **7**A–**7**B illustrate E-model charts that map conditions of a network link to a desired quality of service in accordance with an embodiment.

FIG. **8** illustrates a flow for processing a call request in accordance with an embodiment that utilizes the E-model charts of FIGS. **7**A–**7**B.

FIG. **9** illustrates a telephony communications system that includes a plurality of communities and links between the communities over which call admission control is performed in accordance with an embodiment.

FIGS. **10**A–**10**B illustrate the flow for managing a call request between terminals in different communities of FIG. **9**.

DETAILED DESCRIPTION

In the following description, numerous details are set forth to provide an understanding of the present invention.

However, it will be understood by those skilled in the art that the present invention may be practiced without these details and that numerous variations or modifications from the described embodiments may be possible. For example, although the description refers to telephony communications over data networks, certain aspects of the methods and apparatus described may be advantageously used with other types of communications systems, such as those communicating video or multimedia data (for video conferencing, for example).

Referring to FIGS. 1A and 1B, one embodiment of a telephony communications system **10** includes a number of endpoints or terminals (terminals **14**, **16**, and **30** illustrated) that are capable of performing voice or other audio communications over a packet-based or message-based data network **20**. As used here, "telephony communications" refers to the transmission and receipt of sounds (e.g., voice or other audio signals) between different points in a system using either wireline or wireless links. Example terminals **14**, **16**, and **30** may include computer systems with speech capability, telephone units that include interfaces to the data network **20**, gateways coupled to standard telephones **34** though a public switched telephone network (PSTN) **32**, and other types of communication devices. Telephony communications can occur between any two or more terminals over the data network **20**.

The data network **20** may include, as examples, private networks such as intranets (e.g., local area networks and wide area networks), and public networks such as the Internet. More generally, as used here, a data network is any communications network that utilizes message-based or packet-based communications. In one embodiment, the data network **20** communicates according to the Internet Protocol (IP), as described in Request for Comment (RFC) 791, entitled "Internet Protocol," dated September 1981. The data network **20** may include a single network or link or multiple networks or links that are coupled through gateways, routers, and the like.

In one embodiment, a call server **12** is coupled to the data network **20** to manage telephony communications (e.g., call setup, processing, and termination) between or among the terminals **14**, **16**, and **30** (and other terminals). A policy server **18** may be accessible by the call server **12** to determine usage policy for telephony communications over the data network **20** to control the quality of service on the data network **20**. For example, the policy server **18** may set the telephony usage of the data network **20** for different time periods. During periods of expected high traffic (e.g., business hours), policy server **18** may set a low usage target for telephony communications. On the other hand, during periods of low expected traffic, the target usage of the data network **20** for telephony communications may be set higher.

Additionally, a network monitor system **19** may be coupled to the data network **20** to monitor certain characteristics and conditions of one or more portions of the data network **20**. The characteristics and conditions monitored may include packet delays, jitter, and packet losses. Packet delay refers to a delay experienced in transmission due to high traffic or other conditions. Packet loss refers to the percentage loss of packets. Jitter refers to variations in the delay of different packets in the same transmission. Jitter may contribute to delay on a network link since receiving platforms need to buffer the received data packets to take into account the different delays of packets.

Although only one call server and policy server are illustrated, multiple call servers and policy servers may be

coupled to the data network. In this arrangement, each of the multiple call servers may be responsible for managing call requests from a predetermined group of terminals, and each policy server may be responsible for maintaining usage policy for different portions of the data network **20**. Further, more than one network monitor **19** may be included in the telephony communications system **10**. For example, multiple network monitors may be located to enable monitoring of characteristics and conditions of different portions of the data network **20**. A call server, policy server, and network monitor may be implemented on separate platforms or in the same platform.

To establish a call between two or more terminals for performing telephony communications, a call request is sent from an origination terminal to the call server **12** for processing. The call request includes the IP address of the origination terminal, the directory number of the destination terminal, and a list of one or more resource elements supported by the origination terminal to be used during an established call. A terminal may be relatively busy at the time a call is desired. As a result, processing capability and storage capability in the origination terminal may be limited so that resource elements that require high bandwidth are not indicated as being supported. Examples of resource elements include codecs (coders/decoders), the size of packets carrying audio data, and other resource elements, as explained further below.

By querying the policy server **18**, the call server **12** determines the usage policy for the data network portions over which the call will be established and discards any resource elements that may be inconsistent with that policy. Additionally, the call server **12** can further restrict use of resource elements based on actual usage of transmission resources. Thus, for example, if a relatively large number of calls have been placed through the call server **12**, the types of resource elements that may be employed for further calls may be those that have relatively low bandwidth requirements. Thus, the call server **12** is able to manage call requests based on usage information, including usage policy and/or actual usage of the data network **20**.

Optionally, according to some embodiments, the call server **12** may also query the network monitor **19** to determine the current characteristics and conditions of the network. Selection of resource elements may thus further be based on the current characteristics and conditions of the network (e.g., delays being experienced by packets and percentage of packet loss). Next, the call server **12** ranks the remaining supportable resource elements based on predetermined merit attributes, which may include quality of service, the available bandwidth, expected usage of transmission resources, and other attributes. Selection of the resource elements to use for a particular call is based on the ranking performed by the call server **12**.

One type of resource element is the audio coder/decoder (codec) used by each of the terminals involved in a call session. An audio codec encodes audio signals originating from an audio input device (e.g., microphone) for transmission and decodes received audio data for output to an output device (e.g., a speaker). The codec can be implemented in software. Several types of codecs are available that have varying levels of data compression and data transfer rate requirements. For example, the G.711 codec provides uncompressed communications of voice data, but has a data transfer rate requirement of 64 kbps (kilobits per second) in each direction. Other codecs, such as the G.728, G.729A, G.729, G.723.1, and G.722 have varying compression algorithms and data transfer rate requirements (which are lower

than that of the G.711 codec). The listed G series of audio codecs are recommendations from the International Telecommunication Union (ITU). Other types of codecs may be supported by terminals in further embodiments.

Generally, higher compression leads to a reduced amount of data so that data transfer rate requirements over a link may be reduced. However, because compression of data may cause loss of information, audio quality may be adversely affected. Thus, the two objectives of higher quality audio and lower data transfer rate requirements may conflict.

Conventionally, an origination terminal that desires to establish a voice communication with a destination terminal sends a list of supported codecs to the destination terminal. In response, the destination terminal chooses an acceptable codec from the list. Such a process is provided by the H.323 protocol, which is a recommendation for packet-based multimedia communications systems from the ITU-T. Although such a process allows voice communications employing a commonly supported codec between the origination and destination terminals, it does not take into account the capacity and usage of the link and other transmission resources between the terminals, in this case the data network **20**, as well as other transmission or communications resources.

Another resource element is the network packet size supported by the codec to communicate voice or other audio. As used here, network packet size refers to the size of the network packet used to carry audio data. In one embodiment, the packet includes an IP packet, which includes an IP header and IP payload. In further embodiments, other types of network packets may be employed, depending on the type of the data network **20**. Inside the IP payload may be a TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) packet. A TCP or UDP packet includes a header and payload. For telephony communications, the payload of a UDP packet may include an RTP (Real-Time Transmission Protocol) packet. RTP is a protocol for the transport of real-time data, including audio and video. An RTP packet includes an RTP header and an RTP payload, which can carry one or more frames of audio data. TCP is described in RFC 793, entitled "Transmission Control Protocol," dated September 1981. UDP is described in RFC 768, entitled "User Datagram Protocol," dated August 1980. RTP is described in RFC 1889, entitled "RTP: A Transport Protocol for Real-Time Applications," dated January 1996; and RFC 1890, entitled "RTP Profile for Audio and Video Conference with Minimal Control," dated January 1996.

A frame refers to the duration of a speech sample. For example, a frame may be 10 milliseconds (ms), which indicates that a 10-ms sample of speech is contained in the frame. Other frames include 20 ms, 40 ms, and so forth, samples of speech. Each type of codec can support certain frame sizes. The number of frames that is placed into an RTP payload determines the size of the network packet (e.g., IP packet or other type of packet) used to carry the audio data. During a given call session, the number of frames to be carried in a network packet can be selected. The network packet size has implications on the burden placed on the data network in a given call session. Smaller network packets generally are associated with higher overhead, since more audio data packets are communicated over the data network **20** between terminals. Larger network packets are associated with reduced overhead, but come at the cost of longer delays since a longer speech sample is created between successive transmissions of audio over the data network **20**. Thus, selection of network packet size (as determined by the selection of the number of frames to be carried in the packet) may also lead to a conflict between the objectives of higher quality audio and reduced load on the data network **20** and other transmission resources.

In accordance with some embodiments, a call control mechanism implemented in the terminals, call server(s), policy server(s), and/or network monitor(s) of the telephony communications system **10** balances the need for high audio quality as well as the need to reduce burden on the data network **20** and other transmission resources. The call control mechanism selects a supported codec, network packet size, and/or other resource element that takes into account support for the resource element by all communicating terminals, the available capacity of the data network **20** and other transmission resources, and the objective to achieve the highest possible quality of service. Additional or different criteria may be used in other embodiments.

An origination terminal communicates with the call server **12** over the data network **20** for call control signaling (to set up and terminate a call). After a connection is established between terminals over the data network, the terminals communicate media traffic (voice or other audio) and media traffic signaling with each other through the data network **20**. The call server **12** performs call setup processing, which includes translation of dialed digits (such as 10-digit telephone number) to an IP address of a destination terminal. The call server **12** also keeps tracks of the status (e.g., busy, idle, down, and so forth) of the terminals that it is responsible for. In addition, the call server **12** keeps track of the usage of the transmission facility (the data network **20** and other transmission resources) by the telephony application. As used here, "telephony application" refers to one or more sessions of voice or other audio communications between or among the various terminals.

Referring to the examples of FIGS. **2–5**, processes for establishing a call between an origination terminal (e.g., the terminal **14**, also referred to as terminal **P1**) and a destination terminal (e.g., the terminal **16**, also referred to as terminal **P2**) are illustrated. In the embodiments of FIGS. **2–5**, the call server **12** does not access the network monitor to select resource elements. An embodiment which does is described in connection with FIGS. **7A–7B** and **8**.

FIG. **2** illustrates the messages communicated between the various entities involved in call establishment, and FIG. **3** illustrates the tasks performed by the call server **12** in the call establishment process according to one embodiment. To start a call, the origination terminal **14**, which has an IP address **P1**, sends a call request, such as a Call_Setup message, which is received (at **102**) by the call server **12**. The Call_Setup message includes a number identifying the destination terminal, a codec list including the codecs that are supported by the terminal **14**, a list of supported packet sizes, and/or a list of other supported resource elements. Supported packet sizes are determined by the number of frames and the size of each frame (e.g., 10-ms frame 20-ms frame, and so forth). Example codecs that are supported include G.711, G.728, G.729, G.729A, G.723.1, and G.722 codecs. The G.711 codec communicates uncompressed audio data and requires a 64-kbps data transfer rate, whereas the other codecs provide varying levels of data compression with lower data transfer rate requirements. For example, the G.728 codec requires a 16-kbps transfer rate, the G.729 codec requires an 8-kbps transfer rate, and the G.723.1 codec requires a transfer rate of 6.3 kbps, 5.3 kbps, or less.

In the call server **12**, the list of available codecs and list of supported packet sizes in the Call_Setup message are

received (at **104**) and combined into a candidate list. Alternatively, the different lists of resource elements may be maintained as separate candidate lists.

Based on the Call_Setup message, the call server **12** translates (at **106**) the number (e.g., the dialed number) of the called party into an IP address (e.g., address P2 of the destination terminal **16**). Next, the call server **12** sends (at **107**) a query message to the policy server **18**. The query message includes the IP addresses of the origination and destination terminals (P1, P2) and a request for the usage policy for a telephony application between the pair of terminals at the present time.

The policy server **18** responds to the query by sending a reply message back to the call server **12** to indicate the usage policy for the terminals P1 and P2 for the present call session. The usage policy information may be in the form of predetermined values representing different levels of target usage for telephony communications. Alternatively, the usage policy information may be in the form of information identifying resource elements that are supported or not supported at the present time. Based on the received usage policy information, the call server **12** updates (at **108**) the candidate list by deleting unacceptable codecs. The policy server **18** may set a low usage level for the telephony application because of high traffic carrying traditional data packets (e.g., e-mail traffic, web browsing traffic, file transfer traffic, and so forth). Thus, codecs that have high bandwidth requirements may be deleted (at **108**) from the candidate list. Examples of such high bandwidth codecs include the G.711 codec. In addition, unacceptable packet sizes are also deleted from the candidate list (at **108**), depending on the usage policy. If low usage level is indicated, then shorter packet sizes may be deleted from the list. Thus, the call server **12** selects one or more resource elements (e.g., codecs and packet sizes) that are supported based on the usage policy of the data network **20**.

Optionally, the call server **12** can also perform an additional bandwidth restriction based on the usage of transmission resources. Each connection between a pair of terminals shares a pool of transmission resources (links coupling the terminals that the call server **12** is responsible for, routers and gateways coupling the links, and other resources) with other applications. The call server **12** keeps track of the usage of the pool of transmission resources by tracking the number of voice calls and bandwidth usage. When the usage reaches a predetermined threshold, the call server **12** may further limit the bandwidth usage. The call server **12** may use this limitation to further delete (at **110**) unacceptable codecs, packet sizes, and other resource elements from the candidate list so that a further reduced number of resource elements may be selected.

The call server **12** then sends (at **112**) a query message, e.g., a Query_Resource_Availability message, to the destination terminal P2 to identify the supported codecs, packet sizes, and other resource elements in the destination terminal P2. The results are returned in a Reply_Resource_Availability message, from which the call server **12** can determine the codecs, packet sizes, and other resource elements supported by the destination terminal P2. The candidate list of codecs, packet sizes, and other resource elements is updated based on the available codecs in the destination terminal P2, with unsupported codecs, packet sizes, and other resource elements deleted from the candidate list.

Potentially, all codecs or packet sizes may have been deleted from the candidate list. If either the list of codecs or the list of packet sizes is empty (as determined at **114**), then no supported codec or packet size exists to allow a call to proceed between the terminals P1 and P2, at which point the call server **12** sends (at **116**) a message to terminate the call setup. The call server **12** also informs the origination terminal P1 of the setup failure.

If at least one codec and at least one packet size is available in the candidate list, then the call can proceed. If two or more codecs are present in the candidate list, then the codecs are reordered (at **118**) by applying a merit-based codec ranking algorithm to rank the codecs in the candidate list in the descending merit order (described further below). Packet sizes may also be ordered according to a merit ranking algorithm, as may other resource elements. The codec, packet size, and other resource element having the highest relative rank is selected. Alternatively, selection may be performed by the terminals, which may be adapted to select the highest ranking resource elements from a list.

Next, the call server **12** sends (at **120**) a Call_Setup message to the destination terminal P2, with the Call_Setup message including an identifier of the calling party (either the calling terminal's telephone number or its IP address), the selected codec, packet size, and other resource element. The call server **12** then proceeds (at **122**) to the remaining tasks to be performed in the call setup, including sending a Selected_Resource message identifying the selected codec, packet size, and other resource element back to the origination terminal P1. Alternatively, the codec, packet size, and other resource element may be communicated as parameters in a Setup_Connection message sent by the call server **12** to connect the call between terminals P1 and P2. A media path is then set up between the terminals P1 and P2.

Although reference is made to selection of several resource elements, it is contemplated that further embodiments may select fewer than all the possible types of resource elements in the call management process. For example, call server **12** may perform selection of only codecs to manage bandwidth usage and quality of service on the data network **20**. In addition, if multiple call servers are present in the data network **20**, then communications may occur between call servers to enable selection of resource elements for establishing a call between terminals controlled by the call servers.

Referring further to FIGS. **4** and **5**, another embodiment of performing call establishment in which a codec, packet size, and/or other resource element are selected is illustrated. FIG. **4** illustrates messages exchanged among the entities involved in the call establishment, and FIG. **5** illustrates the tasks performed by the call server **12** in accordance with this embodiment. The tasks performed in the embodiment of FIG. **5** that are common to the tasks performed in the embodiment of FIG. **3** have the same reference numbers. As with the FIGS. **2**–**3** embodiment, the origination terminal P1 sends a Call_Setup message to the call server **12** that includes a list of available codecs, a list of packet sizes, and a list of other resource elements, which are received in a candidate list (at **104**) and updated (at **108**) based on the usage policy in the data network **20** for the telephony application as determined from the policy server **18**. This candidate list may further optionally be updated based on an internal table in the call server **12** on the usage of transmission resources (at **110**). However, instead of querying the destination terminal P2 for its available codecs and supported packet sizes, the call server **12** in this alternative embodiment determines (at **202**) if the candidate list is empty at this point. If so, then a capable codec and packet size have not been found and the call setup is terminated (at

204). However, if the candidate list includes at least one codec and at least one packet size, the call server **12** reorders (at **206**) the codecs and packet sizes (if more than one of each) in the candidate list according to a descending merit score. The candidate list is sent (at **208**) by the caller server **12** to the destination terminal P2 in a Call_Setup message to notify the destination terminal P2 of an incoming call. At this point, the destination terminal P2 may compare the list of its supported codecs to the ones in the candidate list. The destination terminal P2 selects the codec (and other resource elements) having highest relative ranking from the candidate list that is also currently supported by the terminal P2 for the current call. If no capable codec or packet size exists, the destination terminal P2 informs the call server **12** of the rejection. The call server **12** determines (at **210**) if a capable codec and packet size has been identified. If not (as determined from receipt of the rejection message from the destination terminal P2), the call setup is terminated (at **212**).

However, if a capable codec and packet size are identified, the destination terminal P2 informs the call server **12** of the selected codec through a Call_Progress message or some other message. If the call server **12** determines that a capable codec and packet size have been selected, then the call server **12** transmits the selected codec and packet size (at **214**) to the origination terminal P1 in a Setup_Connection message or some other message, such as a Select_Resource message. The call server **12** then proceeds to the remaining tasks to perform for the call setup (at **216**), after which a media path is established between the origination terminal P1 and the termination terminal P2 for voice (or other audio) communications.

Variations of the processes described in connection with FIGS. **2–5** may be performed periodically during a call session between two or more terminals. This allows modification of the selected resource element in response to increases or decreases in the available bandwidth of the data network **20** and other transmission resources, including usage of resources in the terminals themselves.

Referring to FIG. **6**, components of an example terminal and call server are illustrated. In FIG. **6**, the components of the terminal **14**, **16**, or **30** and call server **12** are illustrated. As noted above, the terminal **14**, **16**, or **30** can be one of many types of devices capable of communicating voice over the data network **20**. These terminals may include computer systems, telephones that are configured to communicate over a data network, a gateway system to the public switched telephone network (PSTN), and other communications devices.

The layers of the terminal **14**, **16**, or **30** include a network interface controller **302** that is coupled to the data network **20**. Above the network interface controller **302** is a network device driver **304** and a network stack **306**, such as a TCP/IP or UDP/IP stack. Above the network stack **306** is an RTP layer **308** that performs various tasks associated with real time communications such as telephony communications. Incoming data from the data network **20** is received through the layers **302**, **304**, **306** and **308** and routed to an audio codec **310**, which has been selected from a number of available codecs as discussed above. The incoming data is decoded by the codec **310** and routed to a digital-to-analog (D/A) converter **312** to produce the output at a speaker **314**. Outbound data to the network **20** originates at a microphone **316** or from an application routine **318**. A user can speak into the microphone **316** to communicate voice data over the data network **20**. Alternatively, the application routine **318** (or some other routine) may generate voice or other audio data to be transmitted to the data network **20**. Examples of this may include an automated answering application, such as a voice mail application or a voice prompt application from which users can select to access to various services.

From the microphone, audio signals are passed through an analog-to-digital (A/D) converter **320**, which digitizes the audio signals and passes the digital audio data to the codec **310**. The codec **310** encodes the data and transmits the coded data down layers **308**, **306**, **304**, and **302** to the data network **20**. The audio traffic is communicated through the data network **20** to another terminal to which the terminal **14**, **16**, or **30** has established a call connection.

In addition to the audio traffic path, a control path exists between the terminal **14**, **16**, or **30** and the call server **12** to set up, maintain, and terminate voice calls over the data network **20**. In the terminal **14**, **16**, or **30** one or more application routines **318** may generate control messages that are transmitted to the call server **12** through the network stack **306**, network device driver **304**, network interface controller **302**, and the data network **20**. Control signaling from the call server **12** is similarly received through the same layers from the data network **20** back to the one or more application routines **318**.

In the call server **12**, similar layers may exist. A network interface controller **330** in the call server **12** is coupled to the data network **20**. Above the network interface controller **330** is a network device driver **332** and a network stack **334**, such as a TCP/IP or UDP/IP stack. One or more call processing routines **336** in the call server **12** control the management of calls between terminals that are assigned to the call server **12**. The call processing routines **336** perform the establishment of calls, maintenance of calls, and termination of calls. The call processing routines **336** may also periodically determine the available usage of the data network **20**, which may cause it to update the codec and packet size used by the terminals in the voice communication session over the data network **20**. For example, the call server **12** may maintain a usage table **351** to keep track of the number of active telephony calls and the usage (based on selected resource elements).

In each terminal and call server, various software routines or modules may exist, such as the one or more application routines **318**, network stack **306**, and device driver **304** in the terminal **14**, **16**, or **30** and the one or more call processing routines **336**, network stack **334**, and device driver **332** in the call server **12**. Instructions of such software routines or modules, and others, may be stored in storage units **344** and **349** in the terminal and call server, respectively. The storage units **344** and **349** may include machine-readable storage media including memory devices such as dynamic or static random access memories, erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs), and flash memories; magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; and optical media such as compact discs (CDs) or digital video discs (DVDs).

The instructions may be loaded and executed by control units **340** and **348** in the terminal and call server, respectively, to perform programmed acts. The control units **340** and **348** may include microprocessors, microcontrollers, application-specific integrated circuits (ASICs), programmable gate arrays (PGAs), or other control devices. The terminal **14**, **16**, or **30** may also include a digital signal processor **346** for performing arithmetic intensive operations such as compression and decompression operations performed by the audio codec **310**.

The instructions of the software routines or modules may be loaded or transported into a system or device in one of many different ways. For example, code segments or instructions stored on floppy disks, CD or DVD media, the hard disk, or transported through a network interface card, modem, or other interface mechanism may be loaded into the system or device and executed as corresponding software routines or modules. In the loading or transport process, data signals that are embodied as carrier waves (transmitted over telephone lines, network lines, wireless links, cables, and the like) may communicate the code segments or instructions to the system or device.

The following discusses the merit-based codec ranking in accordance with one embodiment. A modified ranking system may be provided for packet size and/or other resource element selection. The call server **12** maintains a table of characteristics of each codec including the following attributes: voice quality (Q), bandwidth usage (B), and terminal DSP (e.g., digital signal processor **346** in FIG. **6**) resource usage (R). The Q, B, and R attributes may contain numeric values (ranging between 0 and 1). The attribute B in one embodiment may represent the inverse of the actual bandwidth usage, that is, a higher B value indicates low bandwidth usage and a low B value indicates high bandwidth usage. A higher value of R indicates lower consumption of DSP resources. The attribute R similarly represents the inverse of the actual DSP usage. A merit factor M can be computed for each codec in the candidate list as a linear combination of the attributes Q, B, and R according to the following equation:

$$M=W_Q{}^*Q+W_B{}^*B+W_R{}^*R,$$

where W_Q, W_B, and W_R are weights that are assigned to the attributes Q, B, and R, respectively. The values of the weights W_Q, W_B, and W_R may be dynamic and can be based on usage of the pool of transmission resources used for the telephony application. Thus, in one example embodiment, the values of the weights W_Q, W_B, and W_R may be assigned as following:

$$W_Q=(1-t)^*0.8,\ W_B=t,\ \text{and}\ W_R=(1-t)^*0.2,$$

where t is the percentage usage of the pooled transmission resources for the telephony application. The codecs in the candidate list may be arranged in descending order of the merit factor M in one embodiment, from which a codec can be selected for use in the call to be established.

Thus, according to one embodiment, the merit factor M is higher for codecs having relatively high audio quality (Q), low expected bandwidth (e.g., data transfer rate) usage (B), and low expected DSP usage (R). Codecs having relatively low audio quality, high expected bandwidth usage, and high DSP usage will have a lower M value. Thus, generally, the value of the merit factor M is increased with higher audio quality and decreased usage of transmission resources (e.g., links in the data network **20** and DSP **346**).

As noted above, the telephony communication system **10** includes a network monitor **19** for monitoring various characteristics and conditions of one or more portions of the data network **20**. Multiple network monitors may be present for monitoring different portions of the data network **20**. The characteristics and conditions monitored include packet delay, jitter, and percentage of packet loss.

The network monitor **19** may perform monitoring of a network link in a number of different ways. One technique is to use a network monitor having two different nodes on a network link. One node of the network monitor can send test packets targeted to the other node in the network monitor **19**. From the transmission and receipt (or lack of receipt) of test packets, the nodes of the network monitor **19** can determine the delays in transmissions of packets as well as the percentage of packet loss. The network monitor **19** can periodically communicate test packets to monitor the link on a periodic basis. Such a technique may be referred to as a static monitoring technique.

A dynamic technique to monitor a link is to access routers or gateways that communicate with the link. Routers and gateways maintain management information that keep track of delays being experienced with links that the routers and gateways are coupled to as well as amounts of packets that are being lost. Thus, each time a call server accesses a network monitor to request the current characteristics and conditions of a particular link, the network monitor can issue a query to a particular gateway or router to determine the current conditions.

In further embodiments, the network monitor **19** may also provide end-to-end delay and packet loss information based on the several classes of service that may be supported, such as those in a quality-of-service (QOS) enabled network. For example, if the data network **20** employs differential services (Diffserv) to provide QOS, different classes of packets may be defined based on assigned Diffserv code points (DSCPs). One class of packets may include packets delivering voice or other audio data. Other classes may be defined for other types of data that may be communicated through the data network **20**. The different classes of packets may be routed through different queues through network nodes so that higher priority classes of packets are delivered more quickly. The network monitor **19** may track the delays and packet loss information by DSCP, with one DSCP assigned for the voice-over-IP class of service.

Once the packet delay and loss information is determined by the call server **12**, the call server **12** can access a database of models (referred to as E-models) for each call server to determine if a codec can satisfy a desired level of quality based on the prevailing network link conditions. E-models (represented in the form of charts) may also be maintained for the other resource elements. Two E-model charts **350** and **352** are illustrated in FIGS. **7**A and **7**B for the G.729A and G.723.1 codecs, respectively. Each E-model includes a chart mapping packet delays and percentage of packet loss to a desired quality level. In each E-model chart **350** or **352**, an R value represents the desired quality of service. The call server **12** may maintain profiles and policies establishing the desired R-values of calls between different combinations of callers. For example, for internal calls within an organization, a lower quality of service (and therefore lower R value) may be established, whereas external calls are set at higher R values. Other embodiments may use different representations of the quality of audio service of codecs and other resource elements.

In the chart **350** for the G.729A codec, the horizontal axis represents packet delay and the vertical axis represents the R value. The curves **360**A–**360**I represent different percentages of packet losses. In one example, the curve **360**A represents a 0% packet loss, the curve **360**B represents a 0.5% packet loss, the curve **360**C represents a 1% packet loss, the curve **360**D represents a 1.5% packet loss, the curve **360**E represents a 2% packet loss, the curve **360**F represents a 3% packet loss, the curve **360**G represents a 4% packet loss, the curve **360**H represents an 8% packet loss, and the curve **360**I represents a 16% packet loss. Thus, as illustrated in FIG. **7**A, the higher the delay and percentage packet loss, the lower the R value. In one embodiment, an R value of 90

generally indicates that users are very satisfied, an R value of 80 generally indicates that users are satisfied, an R value of 70 generally indicates that some users are dissatisfied, an R value of 60 generally indicates that many users are dissatisfied, and an R value of 50 and below generally indicate that nearly all users are dissatisfied with the level of service.

The chart **352** in FIG. 7B for the G.723.1 codec is similar to the chart **350** in FIG. 7A, with the curves **362A–362I** representing corresponding percentages of packet loss to curves **360A–360I** in FIG. 7A. Thus, given the current packet delay and percentage of packet loss, the charts of the E-models for the various codecs may be accessed to determine which codec can support the desired R value. In further embodiments, different models may be used for codec or other resource element selection.

Thus, referring to FIG. **8**, in accordance with an alternative embodiment that uses E-model charts, such as **350** and **352**, the call server **12** receives (at **370**) a call request from an origination terminal. The call request may identify the resource elements, including codecs, supported by the origination terminal. The call server can perform (at **371**) selection of the codecs and other resource elements based on the usage policy and usage of transmission resources, including the data network **20**, as described above in connection with FIGS. **2–5**. This may reduce the number of codecs and other resource elements.

Further, based on the profiles and policies associated with the identified origination and destination terminals in the call request, the call server identifies (at **372**) the target quality of service (R value). Next, the call server **12** can send (at **374**) query messages to the network monitor **19** to determine the current characteristics and conditions of the network **20**, including network delay and packet loss. Based on the identified delay and packet loss information, the call server **12** accesses (at **376**) the E-model charts of the supported codecs. From the E-model charts, the call server **12** identifies (at **378**) the codecs and other resource elements that satisfy the target R value. Next, the codecs and other resources may be ranked (at **380**) as described above based on various merit attributes to enable selection of one of the codecs and other resource elements to use during the call, as described above.

Some embodiments of the invention may provide one or more of the following advantages. A flexible codec (and other resource element) selection strategy is provided to enforce a policy based on the codec data rate between a pair of terminals where the codec (and other resource element) selection takes into account the capacity and resource limitation of the terminals as well as network traffic load and actual transmission resource usage in each terminal. Selection of resource elements may also be based on the prevailing characteristics and conditions of the network, such as delay and packet loss. Fine policy control over telephony traffic over a data network is made available. Selection may be biased towards high voice quality when traffic is light; however, if other network traffic high, then voice quality may be reduced to reduce the load on the data network.

The codec and other resource element selection technique and apparatus may be used with other applications. For example, for video conferencing communications sessions over a packet-based data network, selection of video codecs may also be used to reduce load on the data network.

Another aspect of managing telephony communications over a data network is call admission control. A call admission procedure determines whether to accept a call request from an origination terminal. If a data network, or any

portion of the data network, has become saturated with traffic (both audio and traditional data packet traffic), then further call requests may be denied to ensure some predetermined quality of service. According to one embodiment of the invention, call admissions is based on usage of links between different groups of terminals (with the groups referred to as communities). Each community includes multiple terminals that are capable of communicating with each other without being subjected to call admissions control. This is made possible by grouping terminals that are coupled to high capacity links, such as LANs. As used here, a community refers to a group of terminals that are coupled by links having relatively high bandwidth. Such terminals may be located geographically close to each other or they may be located over large distances.

Within each community, voice calls between terminals are allowed to proceed when requested. In one embodiment, to provide some limitation on bandwidth usage of the communication link within each community, resource element selection (such as the codec and packet size selection described above) may be used to limit the bandwidth of each call session when large numbers of call sessions are present in the community. In other embodiments, resource selection may be skipped for intra-community calls. The call admission control in some embodiments of the invention is provided for calls made between communities based on usage of the links among the communities.

Referring to FIG. **9**, one arrangement of a voice communication system **400** that includes communities is illustrated. The illustrated multiple links between terminals are logical links, not physical links. The logical links are part of the overall data network, with each link corresponding to a path through the data network between any two terminals. In FIG. **9**, each of the three communities **402**, **404**, and **406** includes its set of terminals. In the community **402**, terminals **408** are coupled to a link **409** (e.g., a LAN, WAN, or other network). A call server **410** is also coupled to the link **409** to manage calls between or among the terminals **408** and between one or more of the terminals **408** and a terminal external to the community **402**. The first community **402** is coupled to the second community **404** over a link. In the second community **404**, terminals **414** are coupled to an internal link **415**. The second community **404** is coupled over another external link **430** to a third community **406**. An internal link **421** in the community **406** is connected to terminals **420**. In the illustrated embodiment, the second and third communities **404** and **406** share a call server **416**, which manages calls within each of, or between, the communities **404** and **406** as well as between a terminal in one of the communities **404** and **406** and another community, such as community **402**. Each server maintains a list of its assigned communities and terminals in each of those communities.

Generally, the links **430** and **432** (and other external links connecting communities) have lower bandwidths than the internal links in each of the communities. However, it is contemplated that exceptions to this exist where an external link may have higher bandwidth than an internal link. For a given community I, the following parameters may be defined: L_I, which represents the limit on a total available bandwidth between the community and a device or system external to the community; M_I, which represents the threshold at which reselection of a codec, packet size, or other resource element is performed to reduce load on a link in a community; N_I, which represents a threshold to restrict outgoing calls; and T_I, which represents the usage of the transmission resources in the community.

Thus, according to one embodiment of a call admission control, outgoing new call requests from the community I may be denied if the value of T_I exceeds the threshold N_I. If the traffic T_I exceeds the threshold M_I, then the call server for the community I can start to perform codec and other resource selection to reduce traffic. Thus, as described above in connection with FIGS. **2–5**, a call server may discard codecs and/or other resource elements based on transmission resources that the call server monitors, including the several thresholds L_I, M_I, and N_I of the community I. In one embodiment, the value of M_I is about 60% to 80% of L_I. The value of N_I can be set at a value closer to or at L_I.

Further, a pair-wise limit can be added for call admission control between communities. In this embodiment, for a given community link between two communities I and J, the following parameters may be defined: L_{IJ}, which represents the limit on a total bandwidth to be used by the community link IJ for the telephony application; M_{IJ}, which represents the threshold at which resource element selection is performed; and T_{IJ}, which represents the usage of transmission resources of the community link IJ. A community link does not have an N parameter since a link has no direction and the concept of incoming or outgoing calls does not apply.

For a terminal in community I to establish a new call with a terminal in community J, the following must be satisfied:

$$T_I<N_I \text{ and } T_{IJ}<L_{IJ}.$$

The first clause essentially states that the traffic between community I and all other communities must be less than the threshold limit N_I. The value of T_I is the sum of all traffic between community I and all other communities, that is

$$T_I = \sum_{allK} T_{IK}.$$

The second clause ($T_{IJ}<L_{IJ}$) specifies that the traffic on the link IJ between communities I and J must be less than the threshold L_{IJ}. If either of the two clauses are not satisfied, then the call request from a terminal in community I is denied. A threshold M_{IJ} is also specified for the link IJ between communities I and J to specify a limit at which resource selection is performed.

The limits L_I, L_{IJ}, M_I, and M_{IJ} may be static (that is, they remain fixed) or adaptive (that is, they may change with changing conditions of the data network). For example, as the data network traffic increases, the threshold values may decrease. The call server can collect statistics regarding the network (such as by accessing a network monitor or other node such as a router or gateway) to determine the conditions of the network. Based on the conditions, e.g., large delays or packet losses, the threshold values may be decreased to maintain high quality of service.

As illustrated in FIG. **9**, the first community **402** has the following parameters: T_1, L_1, M_1, N_1; the second community **404** has the following parameters: T_2, L_2, M_2, and N_2; and the third community **406** has the following parameters: T_3, L_3, M_3, and N_3. The community link **432** has the following parameters: T_{12}, L_{12}, M_{12}; and the community link **430** has the following parameters: T_{23}, L_{23}, and M_{23}.

Referring to FIGS. **10A–10B**, the call admissions control procedure is illustrated for a call between an origination terminal in one community (community I) and a destination terminal in a second community (community J). In the example of FIGS. **10A–10B**, a first call server **500** services community I and a second call server **501** services community J. The call server **500** receives a Call_Setup message from a terminal in community I that includes a list of supported audio codecs and a list of supported packet sizes. The call server **500** then determines (at **502**) whether the call is an intra-community or an inter-community call. If the call is an intra-community call, then the call server **500** in community I performs intra-community call processing and exchanges messages between the terminals involved in the call session (at **504**). Codec and other resource element selection may be performed as described above if the traffic T_I exceeds the threshold value M_I.

If the call is an inter-community call, then the call server **500** determines (at **506**) the name of the origination community, in this case community I. The call server **500** then checks the attributes L_I, M_I, and N_I of the community I. At this point, the call server **500** checks the traffic T_I (between community I and all other communities) against the limit N_I. If T_I exceeds N_I, then the call is denied by the call server **500**. However, if the call request is allowed to proceed, then a candidate list of codecs and packet sizes is then created. Such a list of codecs and packet sizes may be further restricted based on the values of the thresholds L_I, M_I, and N_I. The bandwidth for community I is reserved to reserve capacity for the requested call. This allows the call server **500** to monitor the available bandwidth for further inter-community call requests from terminals in the community I.

A call request message is sent (at **508**) from the call server **500** to the call server **501** that is assigned to community J. The message includes the name of the origination community I as well as the candidate list of codecs and packet sizes. In response to the message from the call server **500**, the call server **501** determines the destination community name J, the community link IJ, and checks the limits L_J, M_J, and N_J, L_{IJ} and M_{IJ} (at **510**). Such a check includes checking if value of T_{IJ} exceeds L_{IJ}. Also, the value of T_J (total traffic of inter-community calls between community J and all other communities) is evaluated against L_J. If T_J exceeds L_J or T_{IJ} exceeds L_{IJ}, then the call is denied and the call server **501** informs the call server **500** of the call termination. The call server **501** may also check T_{IJ} against M_{IJ}, and T_J (total traffic from community J) against M_J, to determine if resource selection is needed.

From the limits, the call server **501** may further restrict the list of allowed codecs and packet sizes. Bandwidth is then reserved for the community J and link IJ for the requested call. The call server **501** then sends a Call_Setup message (at **512**) to the destination terminal in community J. The Call_Setup message includes the codec and packet size candidate list. In response to the Call_Setup message, the destination terminal sends back a Call_Connect message (at **514**) that identifies a selected codec and a packet size. The call server **501** and destination terminal may select a codec and packet size using techniques described in connection with FIGS. **2–5**, which uses a ranking algorithm. Based on the returned Call_Connect message identifying the selected codec and packet size, the call server **501** corrects (at **516**) the expected bandwidth usage of community I and link IJ. The call server **501** then sends back (at **518**) a Connect/Answer message to the call server **500** that includes an identification of the termination community link (J) and the selected codec and packet size. Based on the identification of the selected codec and packet size, the expected bandwidth usage in the community I for the call session is corrected, and the expected bandwidth usage of the community link IJ is updated (at **520**).

At this point, a call has been connected between the origination terminal and the destination terminal in commu-

nities I and J, respectively. If the origination terminal desires to terminate the phone call, then it sends a release message (at **522**) to the call server **500**. In response, the call server **501** updates (at **530**) its bandwidth usage of community I and link IJ and sends a release message (at **524**) to the call server **501**. In response to the release message, the call server **501** updates (at **526**) the bandwidth usage for community J and link IJ to reflect termination of the call. The call server **501** sends a release complete message (at **528**) to the destination call server to terminate the call.

In some cases, it is easy to determine whether or not a call is intra-community or inter-community. For example, the translation component in a call server can be equipped with information indicating whether or not the call request is for an intra-community call. However, in some other cases, the origination terminal's call server cannot accurately determine if a call request is for an intra- or inter-community call. For example, although a call request may identify a destination terminal in another community, the destination terminal may have forwarded the call back to a terminal in the origination community. In this case, the origination terminal's call server may assume that the call is an inter-community call and delete any codecs and other resource elements from the candidate list based on the origination terminal's community threshold values. However, the call may still be determined to be an intra-community call by a second call server associated with the assumed destination terminal. The second call server may determine that the call has been forwarded back to the community of the origination terminal. Thus, in an embodiment in which intra-community calls are not subject to resource element selection, the call request should not be denied even if the resulting candidate list is empty since the call may be forwarded back to the origination terminal's community for an intra-community call. However, if the call is indeed inter-community, and the candidate codec list is empty, then the call is denied by the second call server.

A call management method and apparatus has been described to offer call admissions control and selection of resource elements to more effectively manage usage of a data network for telephony communications while providing a higher quality of service.

While the invention has been disclosed with respect to a limited number of embodiments, those skilled in the art will appreciate numerous modifications and variations therefrom. It is intended that the appended claims cover all such modifications and variations as fall within the true spirit and scope of the invention.

What is claimed is:

1. A method of managing calls over a data network, comprising:

determining usage information of the data network;

receiving a call request for establishing a call between at least two network terminals; and

selecting one or more of a plurality of resource elements as candidates for use in the requested call in response to the call request based on usage information of the data network, wherein the resource elements define one or more characteristics of data exchanged between the network terminals,

wherein the selecting includes selecting one or more resource elements based on usage policy set by a policy server.

2. A method of managing calls over a data network, comprising:

determining usage information of the data network;

receiving a call request for establishing a call between at least two network terminals;

selecting one or more of a plurality of resource elements as candidates for use in the requested call in response to the call request based on usage information of the data network, wherein the resource elements define one or more characteristics of data exchanged between the network terminals;

receiving information relating to the plurality of resource elements during establishing of the call; and

selecting one or more of the plurality of resource elements based on support for the one or more resource elements in each of the at least two network terminals.

3. A method of managing calls over a data network, comprising:

determining usage information of the data network;

receiving a call request for establishing a call between at least two network terminals;

selecting one or more of a plurality of resource elements as candidates for use in the requested call in response to the call request based on usage information of the data network, wherein the resource elements define one or more characteristics of data exchanged between the network terminals; and

ranking the resource elements according to merit based on quality of the requested call and expected bandwidth usage of the data network.

4. The method of claim **3**, wherein the ranking of the resource elements is further based on expected usage of a digital signal processing element of each terminal.

5. A method of managing calls over a data network, comprising:

determining usage information of the data network;

receiving a call request for establishing a call between at least two network terminals;

selecting one or more of a plurality of resource elements as candidates for use in the requested call in response to the call request based on usage information of the data network, wherein the resource elements define one or more characteristics of data exchanged between the network terminals; and

performing call admissions control to accept or deny the call request,

wherein the at least two terminals are defined in at least two communities coupled by a link, and wherein performing call admissions control includes performing call admissions control based on a threshold set for the link between the communities.

6. The method of claim **5**, wherein performing call admissions control is based on usage of a link in the data network between groups of terminals.

7. The method of claim **5**, further comprising bypassing the call admissions control for an intra-community call within each community.

8. A server for managing calls in a system having a network, comprising:

an interface to the network to receive a call request to establish a call between two endpoints on the network; and

a control unit adapted to process the call request and to control selection of one or more of a plurality of resource elements as candidates to be employed by the endpoints in the call based on usage of the network,

wherein the resource elements comprise at least one of codecs to be employed by the endpoints in the call and sizes of messages to be used for carrying audio data in the call,

wherein the control unit is adapted to rank the resource elements by one or more predetermined criteria,

wherein the control unit is adapted to present the ranked resource elements to at least one of the endpoints for the at least one endpoint to select a resource element.

9. The server of claim **8**, wherein the control unit is adapted to select the resource element having a highest relative rank.

10. The server of claim **9**, wherein the control unit is adapted to determine resource elements supported by the endpoints.

11. An article including one or more machine-readable storage media containing instructions to manage calls within a telephony system, the instructions when executed causing a controller to:

receive a call request containing information identifying an origination endpoint, a destination endpoint, and one or more resource elements supported by the origination endpoint;

select one or more of the one or more resource elements based on perceived audio quality and usage of a data network;

present the selected one or more resource elements as available for use in a call between endpoints; and

receive information relating to the one or more resource elements during call establishment.

12. A method of managing calls in a telephony system, comprising:

defining a plurality of communities each including one or more communication endpoints;

assigning at least first and second usage threshold values to a link between communities; and

processing a call request based on the usage threshold values,

wherein the processing includes determining whether to admit the call request over the link based on the first usage threshold value,

wherein the processing further includes selecting one or more resource elements to be used during a call session between endpoints over the link based on the second usage threshold value,

wherein the processing includes admitting the call request over the link and performing selecting of the resource elements if usage over the link exceeds the second usage threshold value but is less than the first usage threshold value.

13. The method of claim **12**, wherein the assigning includes assigning a threshold value indicating the available bandwidth on the link between the communities.

14. The method of claim **12**, wherein the assigning includes assigning a usage threshold value over which further outgoing calls from a community is prohibited.

15. A call establishment method, comprising:

determining a candidate list of coding resource members associated with a call request;

checking a usage policy for the call request;

removing from the candidate list a first set of coding resource members whose bandwidth requirements exceed the usage policy;

ranking a second set of coding resource members of the candidate list according to merit, the second set being distinct from the first set;

selecting from the second set a coding resource member having a highest relative merit; and

establishing a call specified by the call request using the selected coding resource member.

16. The call establishment method of claim **15**, wherein the determining includes:

receiving at least one supported coding resource of an endpoint specified with the call request; and

assembling the candidate list from the at least one received supported coding resource.

17. The call establishment method of claim **16**, wherein the call request specifies an originating endpoint and at least one destination endpoint; and

wherein the receiving comprises receiving at least one supported coding resource member for each of the originating and at least one destination endpoints.

18. The call establishment of claim **15**, wherein the ranking comprises ranking the second set of coding resource members according to at least one of perceived voice quality, bandwidth usage, and endpoint digital signal processing resource usage.

19. The call establishment method of claim **15**, wherein the call establishing fails to establish the call when the second set is empty.

20. A method of managing calls over a data network, comprising:

determining usage information of the data network;

receiving a call request for establishing a telephony communications session between at least two network terminals;

selecting one or more of a plurality of resource elements as candidates for use in the requested telephony communications session in response to the call request based on usage information of the data network, wherein the resource elements define one or more characteristics of data exchanged between the network terminals; and

receiving information relating to the plurality of resource elements during establishing of the telephony communications session.

21. The method of claim **20**, wherein the selecting includes selecting one or more resource elements based on actual usage of the data network.

22. The method of claim **20**, wherein the selecting includes selecting one or more codecs as candidates for use in each network terminal.

23. The method of claim **20**, wherein the selecting includes selecting one or more sizes of a packet as candidates for carrying audio data in the requested telephony communications session.

24. The method of claim **20**, further comprising:

determining a condition of the data network, wherein the selecting is further based on the determined condition.

25. The method of claim **24**, wherein the determining includes determining a delay in the transmission of packets in the data network.

26. The method of claim **24**, wherein the determining includes determining a percentage of packet loss in the data network.

27. The method of claim **24**, further comprising determining an expected quality of service based on the determined condition of the data network.

28. A server for managing calls in a system having a network, comprising:

an interface to the network to receive a call request to establish a call between two endpoints on the network; and

a control unit adapted to process the call request and to control selection of one or more of a plurality of resource elements as candidates to be employed by the endpoints in the call based on usage of the data network, wherein the resource elements comprise at least one of codecs to be employed by the endpoints in the call and sizes of messages to be used for carrying audio data in the call,

wherein the control unit is adapted to receive information relating to the plurality of resource elements from an originating one of the two endpoints during call establishment.

29. The server of claim **28**, wherein the control unit is adapted to retrieve information regarding usage of the network, the control unit controlling selection of the one resource element based on the usage.

30. The server of claim **28**, wherein the sizes of messages are determined by a selected number of frames carrying audio data in each message.

31. The server of claim **28**, wherein the calls include telephony calls.

32. The server of claim **28**, wherein the control unit is adapted to receive the information in the call request.

* * * * *

EXHIBIT K

US005991389A

United States Patent [19]
Ram et al.

[11] **Patent Number: 5,991,389**
[45] **Date of Patent: Nov. 23, 1999**

[54] **PROGRAMMABLE SERVICE ARCHITECTURE FOR CALL CONTROL PROCESSING**

[75] Inventors: **Geetha R. Ram**, Plano; **Kent W. Smith**, Allen; **Keith W. Landau**, Plano; **Roger L. Thompson**, McKinney, all of Tex.

[73] Assignee: **Northern Telecom Limited**, Quebec, Canada

[21] Appl. No.: **08/866,229**

[22] Filed: **May 30, 1997**

Related U.S. Application Data

[60] Provisional application No. 60/020,713, Jun. 13, 1996.

[51] **Int. Cl.**[6] **H04M 3/42**
[52] **U.S. Cl.** **379/230**; 379/207; 379/242
[58] **Field of Search** 379/207, 219, 379/220, 221, 229, 230, 242, 243, 244, 245, 88.22

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,878,240	10/1989	Lin et al.	379/88.22
5,418,844	5/1995	Morrisey et al.	379/207
5,426,694	6/1995	Hebert	379/242
5,517,563	5/1996	Norell	379/230
5,541,917	7/1996	Farris	379/230
5,546,453	8/1996	Hebert	379/242
5,553,127	9/1996	Norell	379/207
5,586,177	12/1996	Farris et al.	379/207
5,590,188	12/1996	Crockett	379/207
5,602,909	2/1997	Carkner et al.	379/207
5,610,977	3/1997	Williams et al.	379/207
5,822,422	10/1998	Daase et al.	379/242

FOREIGN PATENT DOCUMENTS

0505092	9/1992	European Pat. Off. .
0620693	10/1994	European Pat. Off. .
2240905	8/1991	United Kingdom .
95/23483	8/1995	United Kingdom .
93/20639	10/1993	WIPO .
94/06252	3/1994	WIPO .
95/32577	11/1995	WIPO .

OTHER PUBLICATIONS

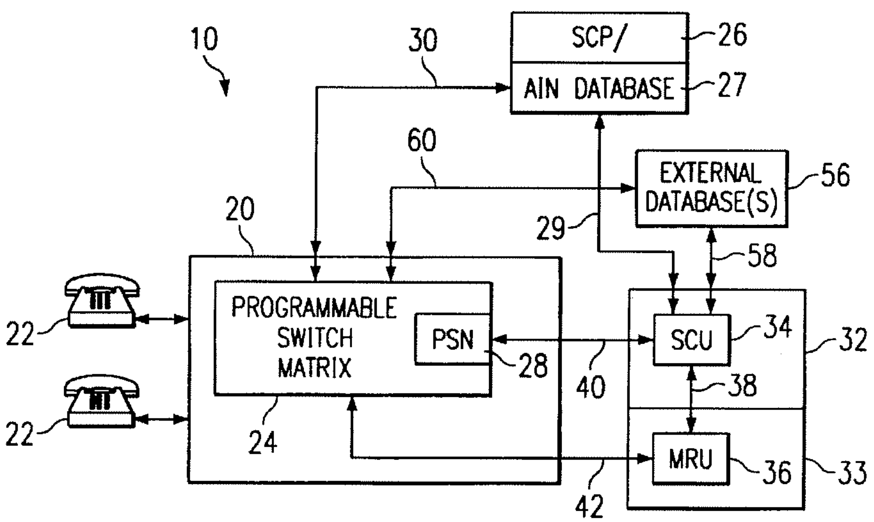
Yang et al, "The Design and Implementation of a Service Logic Execution Environment Platform," Globecom 93, vol. 3, Nov. 29, 1993, pp. 1911–1917.

Primary Examiner—Scott Wolinsky
Attorney, Agent, or Firm—Carr & Storm, L.L.P.

[57] **ABSTRACT**

An apparatus and method for providing a telephone operating company with the ability to rapidly deploy advanced services into a public switched telephone network includes a programmable switch matrix, a service control unit (SCU), and a media resource unit (MRU). The call processing of a call (associated with one or more ports on the programmable switch matrix) is controlled externally by the SCU when particular triggering criteria is met (i.e., the call requires or desires control by the SCU). Call control processing is achieved through a high-speed communications link between the programmable switch matrix and the SCU using a communications protocol defining a comprehensive set of primitives (instructions) for call manipulation and control at the programmable switch matrix. The SCU executes different service application software programs that operate within the SCU for different types of service calls that are under the control of the SCU. The MRU interconnects between the SCU and the programmable switch matrix to provide voice processing and message capabilities for connection to a service call via the programmable switch matrix.

27 Claims, 17 Drawing Sheets



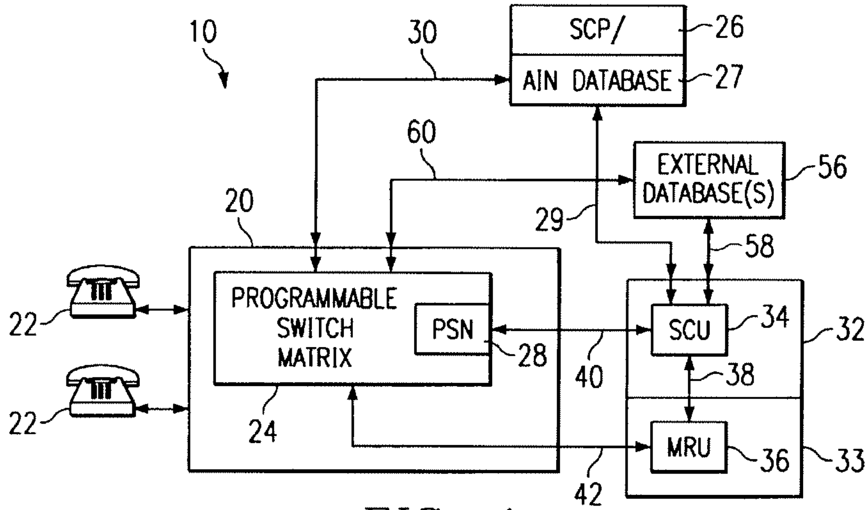

10
30
SCP/
AIN DATABASE
26
27
60
EXTERNAL
DATABASE(S)
56
20
29
58
PROGRAMMABLE
SWITCH
MATRIX
PSN
28
SCU
34
32
22
38
40
MRU
36
33
24
42

FIG. 1

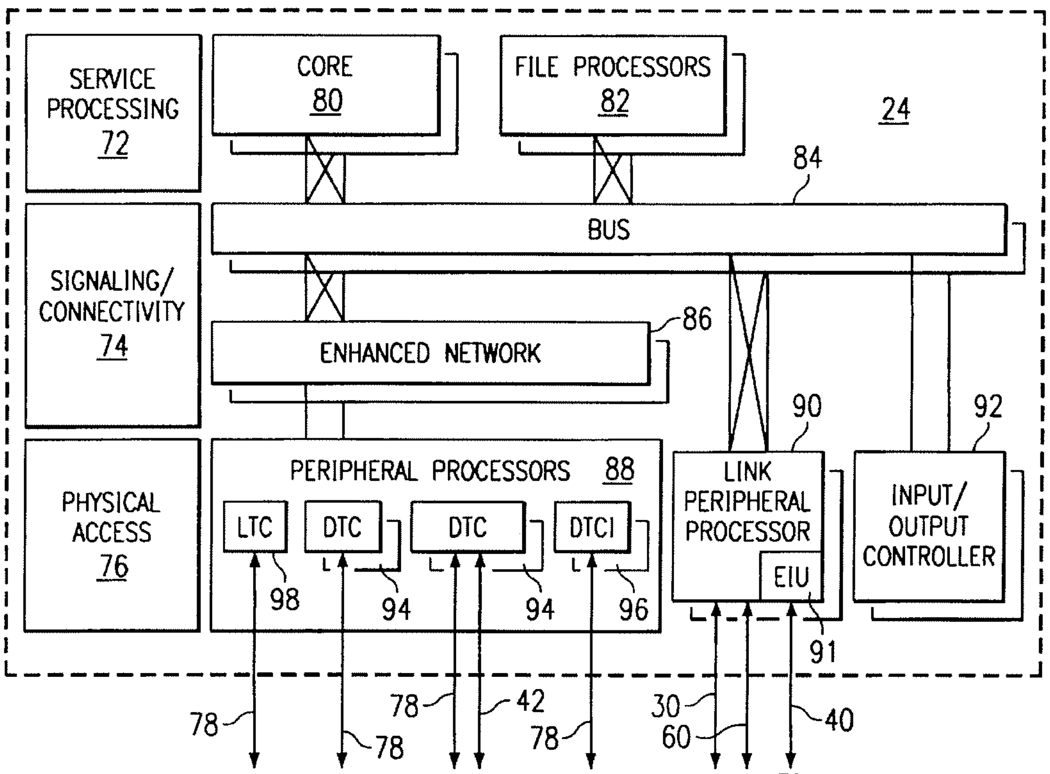

SERVICE
PROCESSING
72
CORE
80
FILE PROCESSORS
82
24
84
BUS
SIGNALING/
CONNECTIVITY
74
ENHANCED NETWORK
86
90
92
PHYSICAL
ACCESS
76
PERIPHERAL PROCESSORS 88
LTC
DTC
DTC
DTCI
LINK
PERIPHERAL
PROCESSOR
EIU
INPUT/
OUTPUT
CONTROLLER
98
94
94
96
91
78
78
42
78
78
30
60
40
TO
SCU

FIG. 2

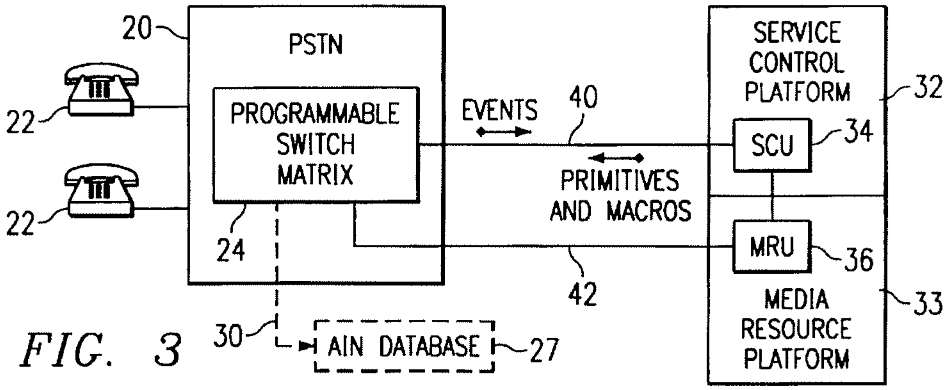


FIG. 3

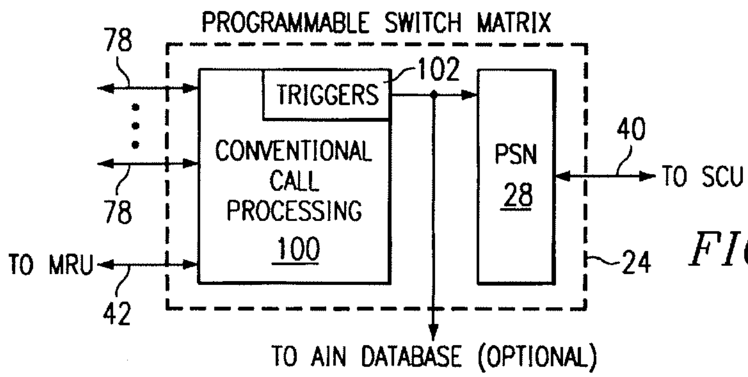


FIG. 4

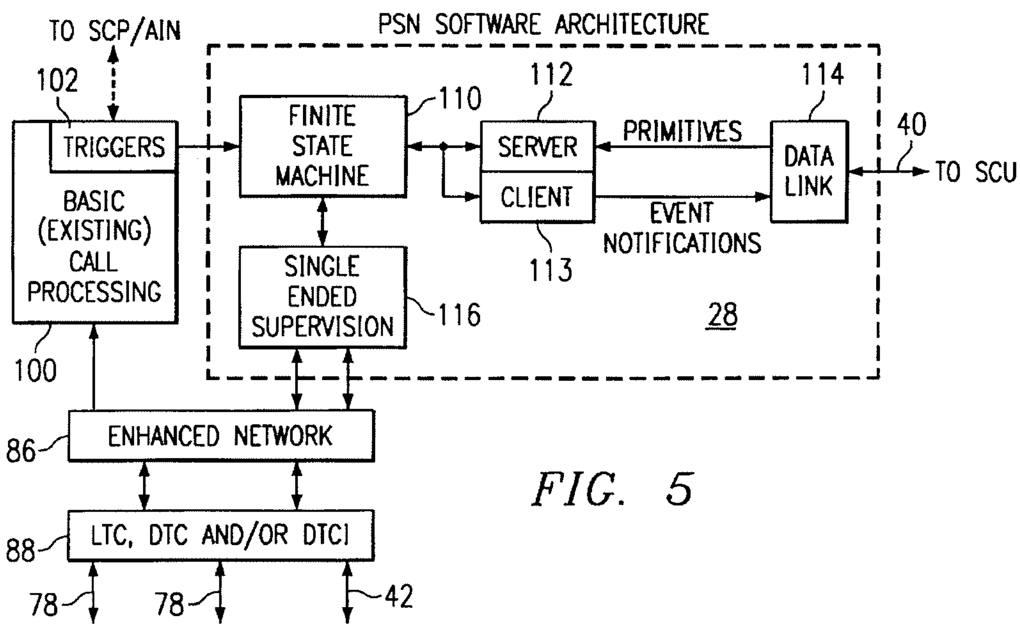
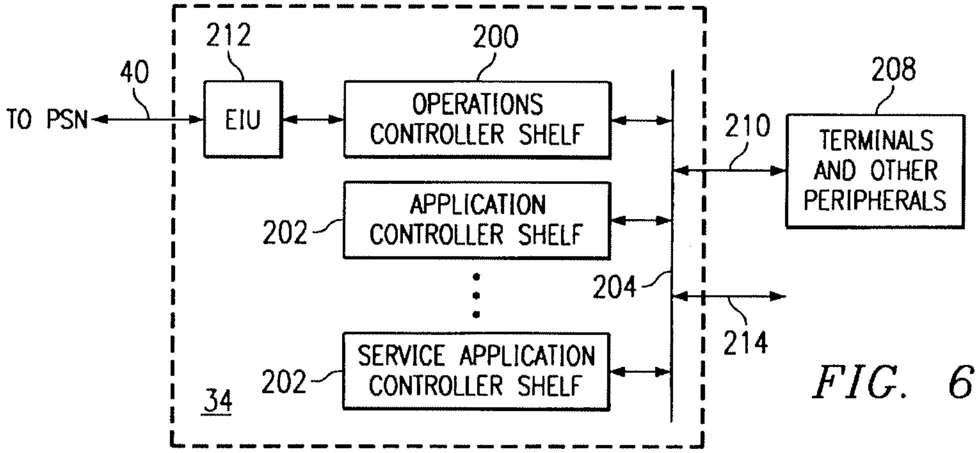
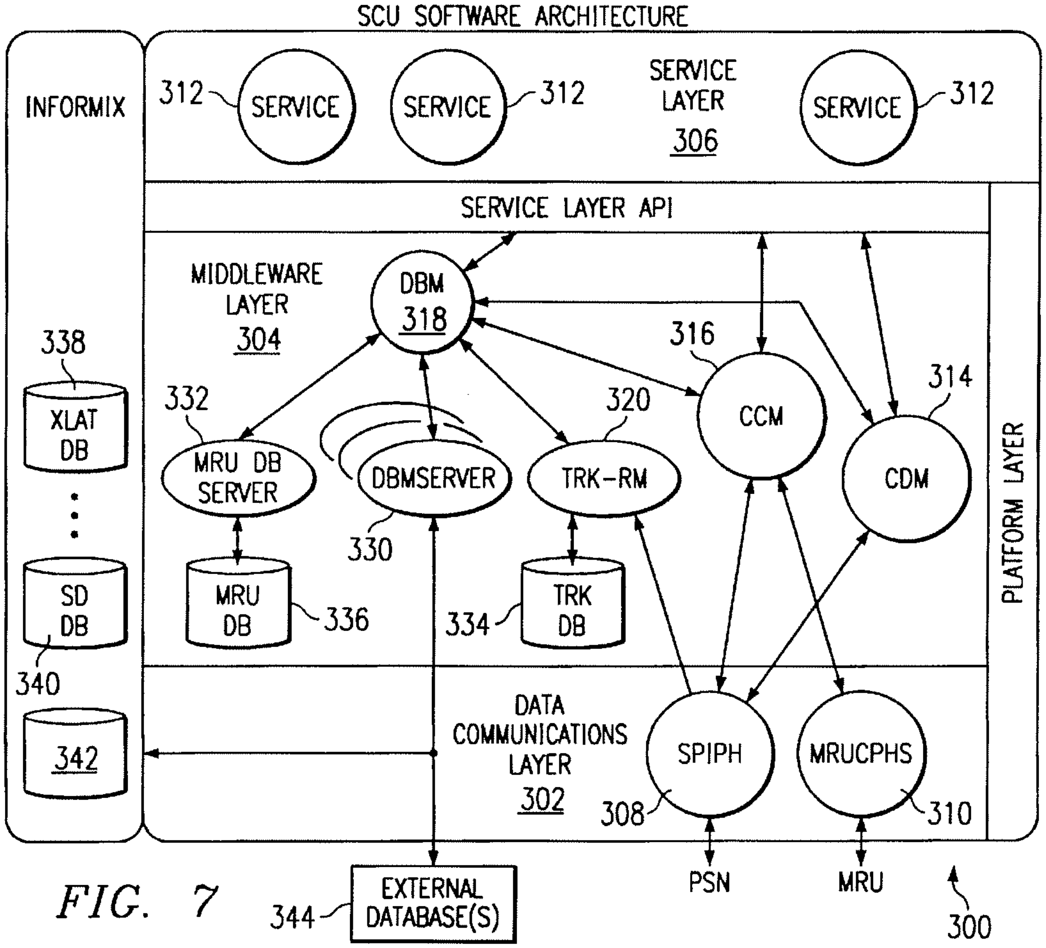


FIG. 5



212
200
40
TO PSN
EIU
OPERATIONS CONTROLLER SHELF
208
210
TERMINALS AND OTHER PERIPHERALS
APPLICATION CONTROLLER SHELF
202
204
214
SERVICE APPLICATION CONTROLLER SHELF
202
34

FIG. 6



SCU SOFTWARE ARCHITECTURE
INFORMIX
312
SERVICE
SERVICE
312
SERVICE LAYER
306
SERVICE
312
SERVICE LAYER API
MIDDLEWARE LAYER
304
DBM
318
316
338
XLAT DB
332
320
314
CCM
CDM
MRU DB SERVER
DBMSERVER
TRK-RM
PLATFORM LAYER
330
SD DB
MRU DB
336
334
TRK DB
340
DATA COMMUNICATIONS LAYER
302
SPIPH
MRUCPHS
342
308
310
PSN
MRU
344
EXTERNAL DATABASE(S)
300

FIG. 7

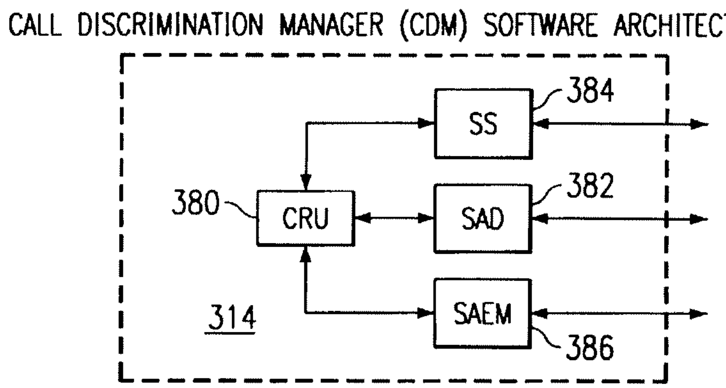

CALL DISCRIMINATION MANAGER (CDM) SOFTWARE ARCHITECTURE
SS
384
380
CRU
SAD
382
314
SAEM
386

FIG. 8

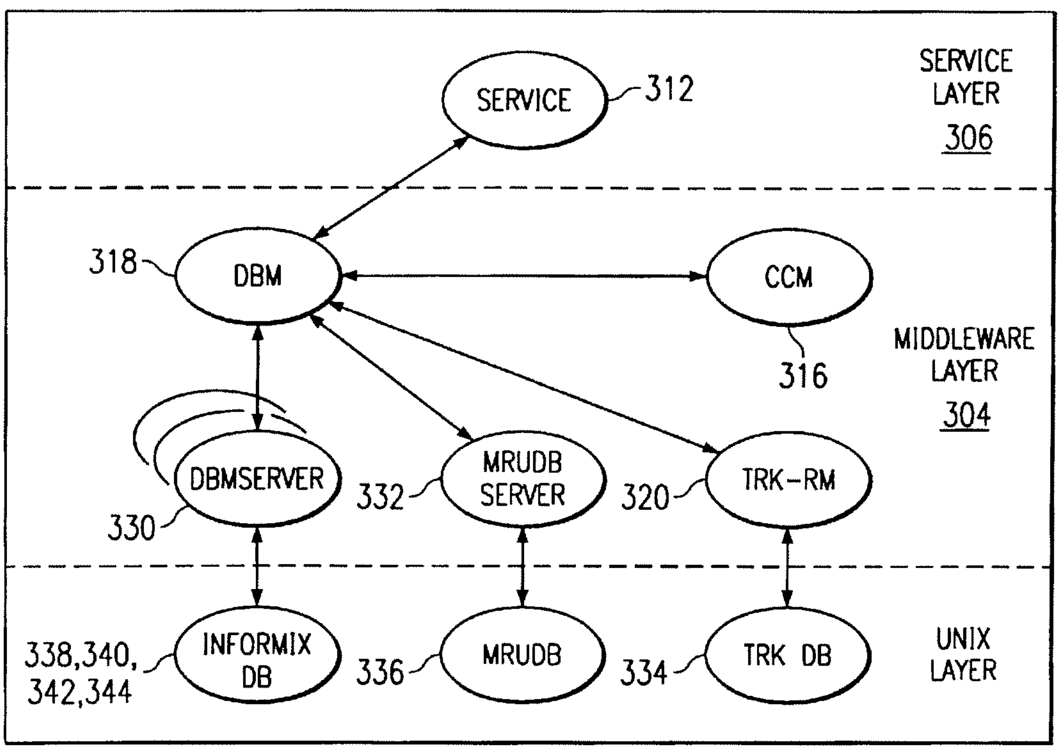

SERVICE
312
SERVICE LAYER
306
318
DBM
CCM
316
MIDDLEWARE LAYER
304
DBMSERVER
330
332
MRUDB SERVER
320
TRK-RM
338,340, 342,344
INFORMIX DB
336
MRUDB
334
TRK DB
UNIX LAYER

FIG. 9

PACKAGING OF PRIMITIVES/MACROS (SCU TO PSN)

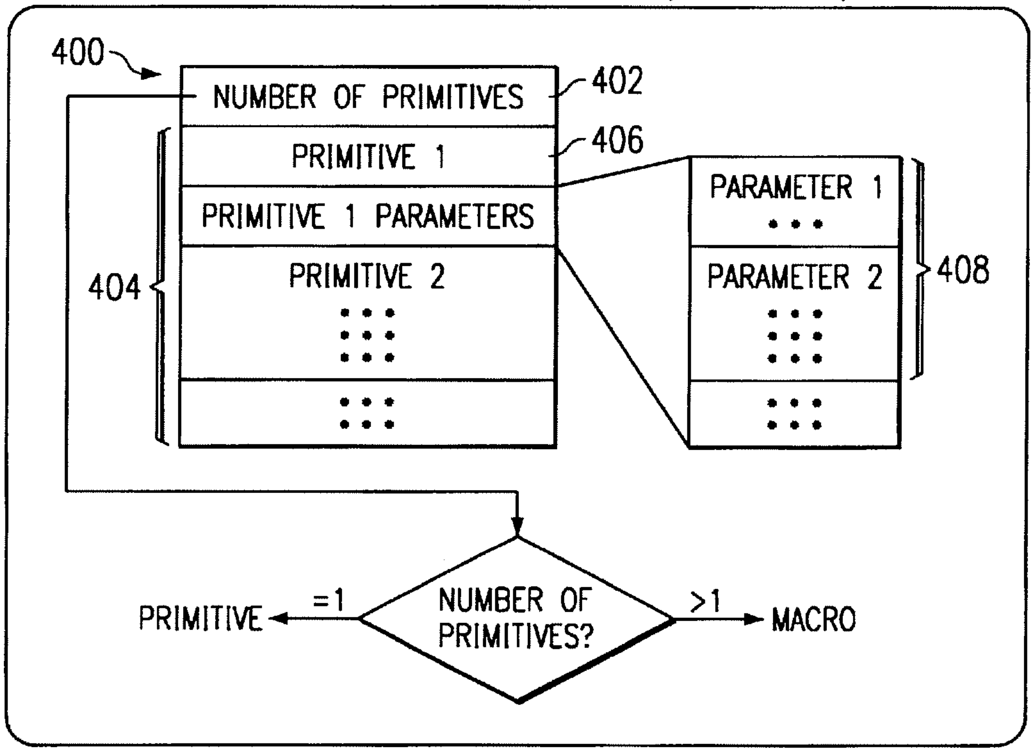


FIG. 10

PACKAGING OF EVENT NOTIFICATIONS (PSN TO SCU)

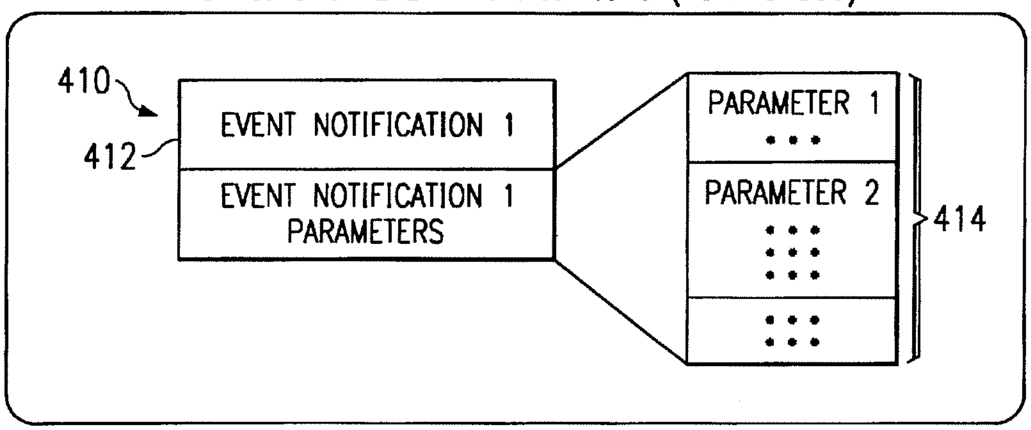
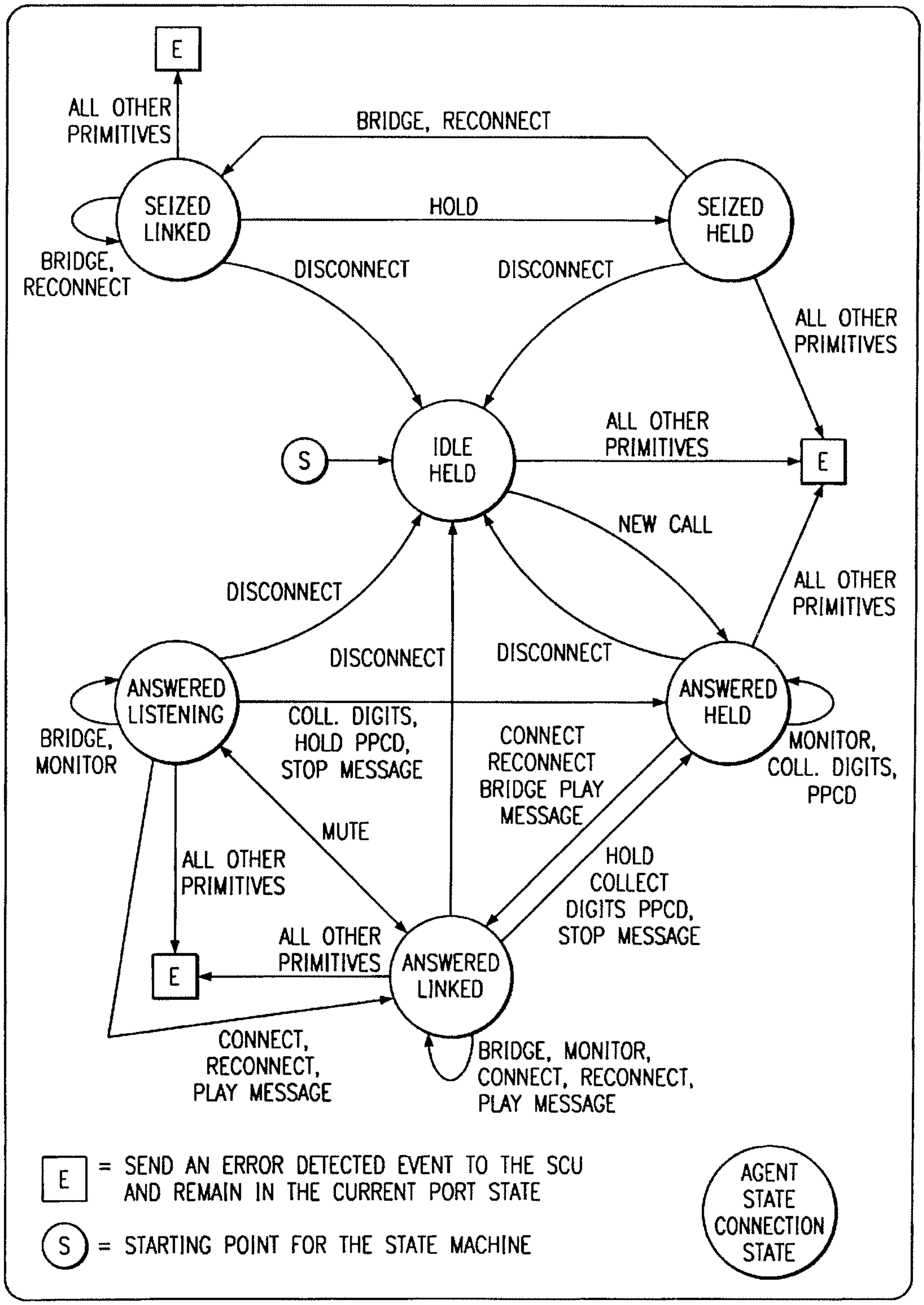


FIG. 11



E
ALL OTHER PRIMITIVES
BRIDGE, RECONNECT
SEIZED LINKED
HOLD
SEIZED HELD
BRIDGE, RECONNECT
DISCONNECT
DISCONNECT
ALL OTHER PRIMITIVES
S
IDLE HELD
ALL OTHER PRIMITIVES
E
NEW CALL
DISCONNECT
ALL OTHER PRIMITIVES
DISCONNECT
DISCONNECT
ANSWERED LISTENING
ANSWERED HELD
BRIDGE, MONITOR
COLL. DIGITS, HOLD PPCD, STOP MESSAGE
CONNECT RECONNECT BRIDGE PLAY MESSAGE
MONITOR, COLL. DIGITS, PPCD
MUTE
ALL OTHER PRIMITIVES
HOLD COLLECT DIGITS PPCD, STOP MESSAGE
ALL OTHER PRIMITIVES
E
ANSWERED LINKED
CONNECT, RECONNECT, PLAY MESSAGE
BRIDGE, MONITOR, CONNECT, RECONNECT, PLAY MESSAGE
E = SEND AN ERROR DETECTED EVENT TO THE SCU AND REMAIN IN THE CURRENT PORT STATE
S = STARTING POINT FOR THE STATE MACHINE
AGENT STATE CONNECTION STATE

FIG. 12a

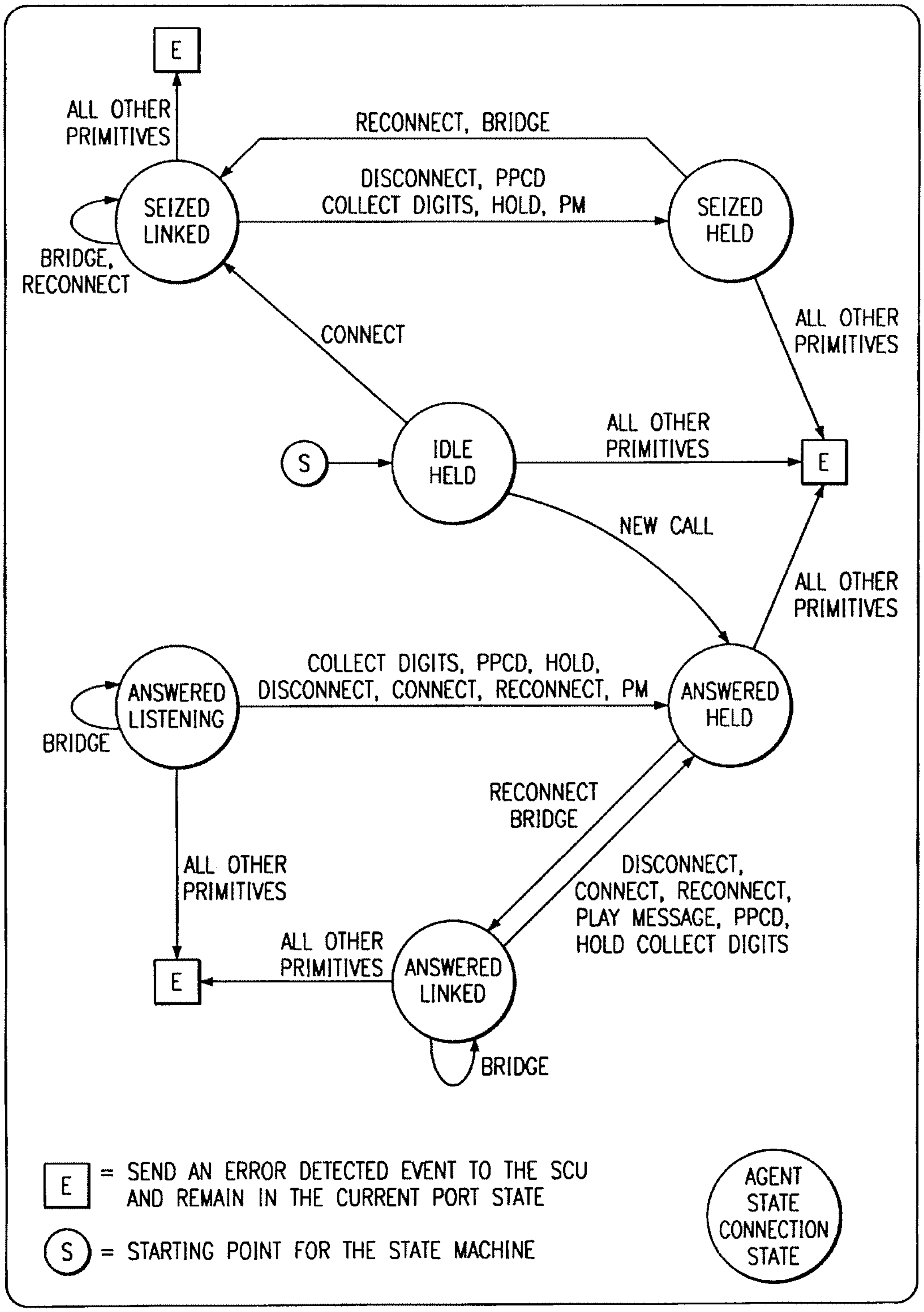
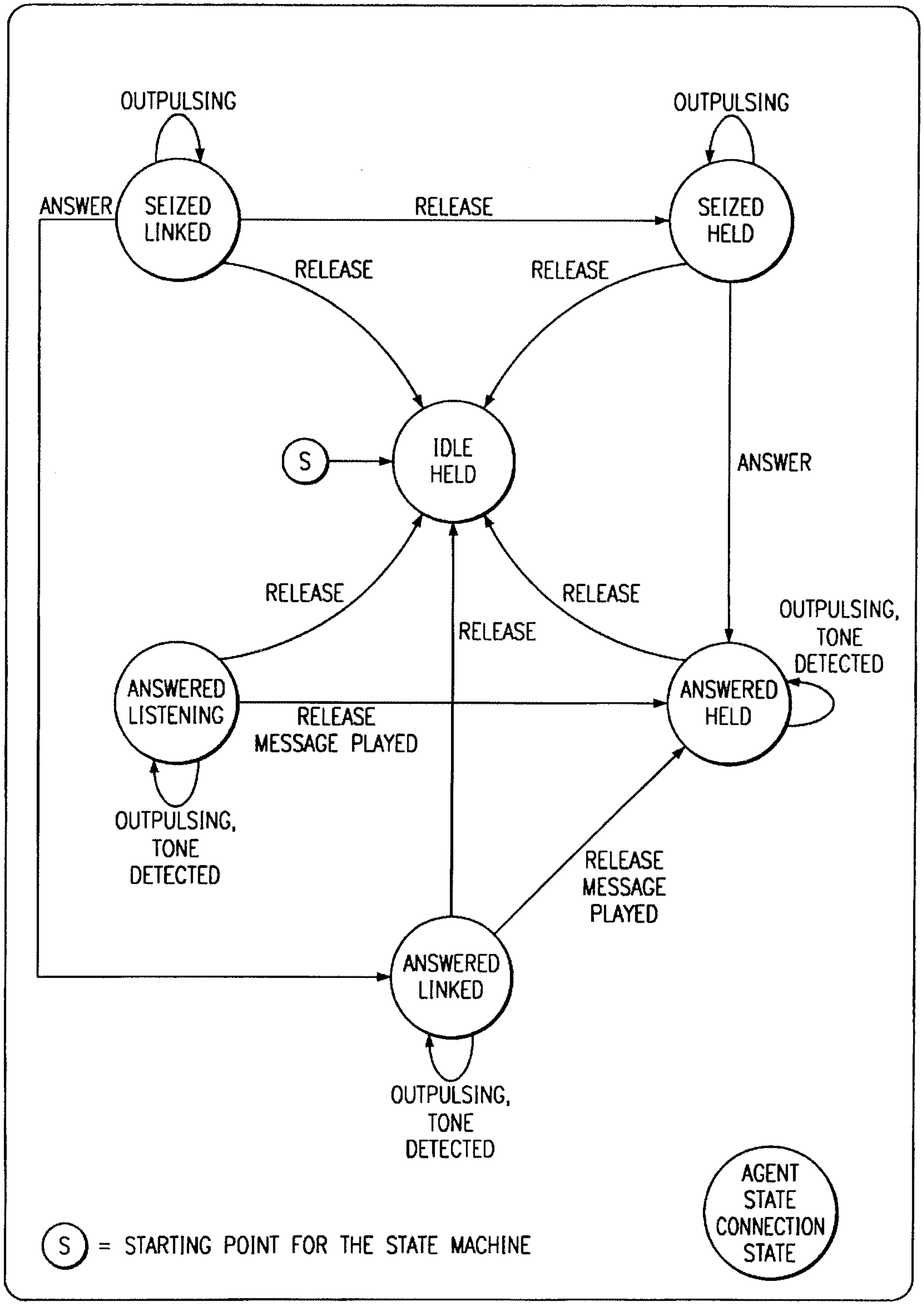

E
ALL OTHER PRIMITIVES
RECONNECT, BRIDGE
DISCONNECT, PPCD
COLLECT DIGITS, HOLD, PM
SEIZED LINKED
SEIZED HELD
BRIDGE, RECONNECT
CONNECT
ALL OTHER PRIMITIVES
S
IDLE HELD
ALL OTHER PRIMITIVES
E
NEW CALL
ALL OTHER PRIMITIVES
COLLECT DIGITS, PPCD, HOLD,
DISCONNECT, CONNECT, RECONNECT, PM
ANSWERED LISTENING
ANSWERED HELD
BRIDGE
RECONNECT BRIDGE
ALL OTHER PRIMITIVES
DISCONNECT,
CONNECT, RECONNECT,
PLAY MESSAGE, PPCD,
HOLD COLLECT DIGITS
ALL OTHER PRIMITIVES
E
ANSWERED LINKED
BRIDGE
E = SEND AN ERROR DETECTED EVENT TO THE SCU AND REMAIN IN THE CURRENT PORT STATE
S = STARTING POINT FOR THE STATE MACHINE
AGENT STATE CONNECTION STATE

FIG. 12b



OUTPULSING
OUTPULSING
ANSWER
SEIZED LINKED
RELEASE
SEIZED HELD
RELEASE
RELEASE
S
IDLE HELD
ANSWER
RELEASE
RELEASE
RELEASE
OUTPULSING, TONE DETECTED
ANSWERED LISTENING
RELEASE MESSAGE PLAYED
ANSWERED HELD
OUTPULSING, TONE DETECTED
RELEASE MESSAGE PLAYED
ANSWERED LINKED
OUTPULSING, TONE DETECTED
S = STARTING POINT FOR THE STATE MACHINE
AGENT STATE CONNECTION STATE

FIG. 12c

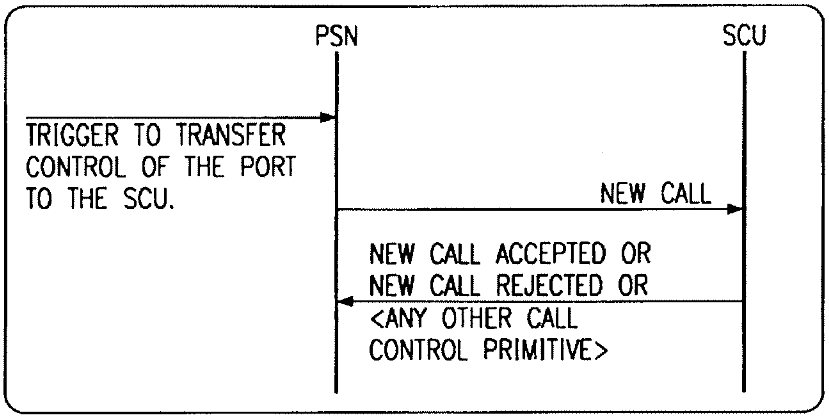


FIG. 13

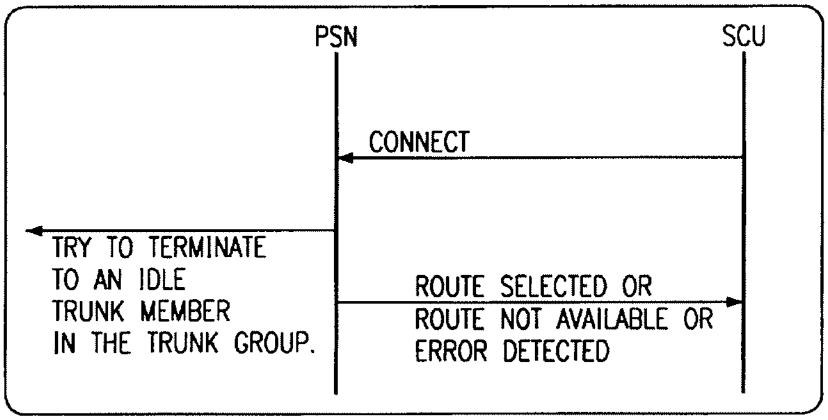


FIG. 14

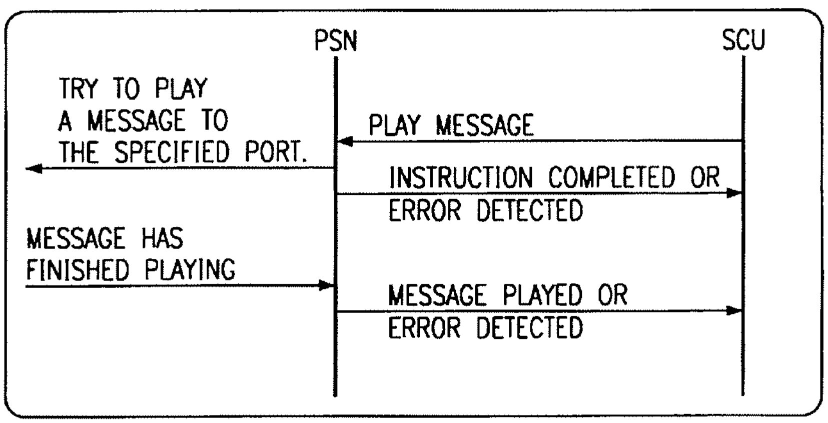


FIG. 15

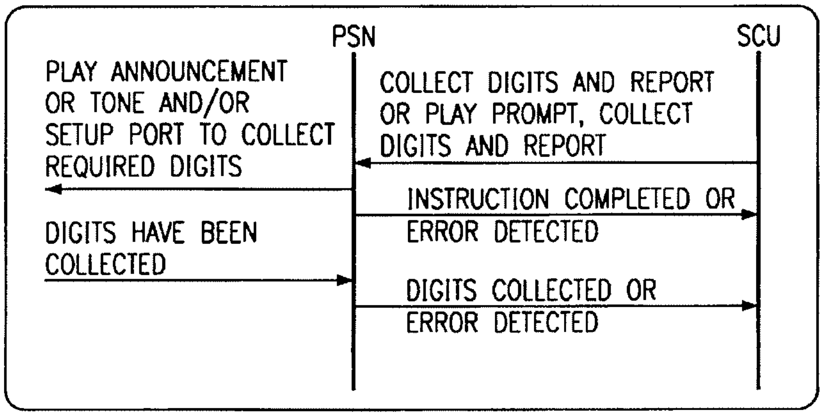


FIG. 16

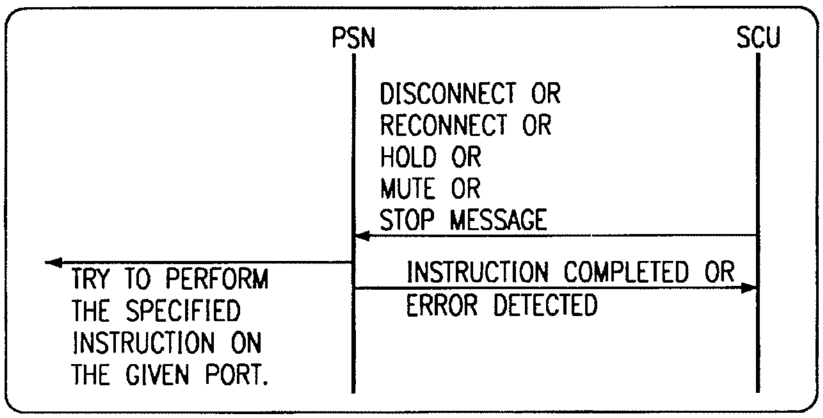


FIG. 17

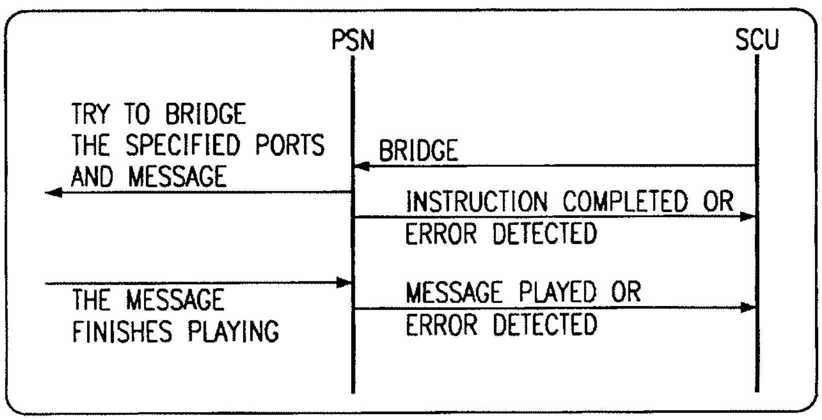


FIG. 18

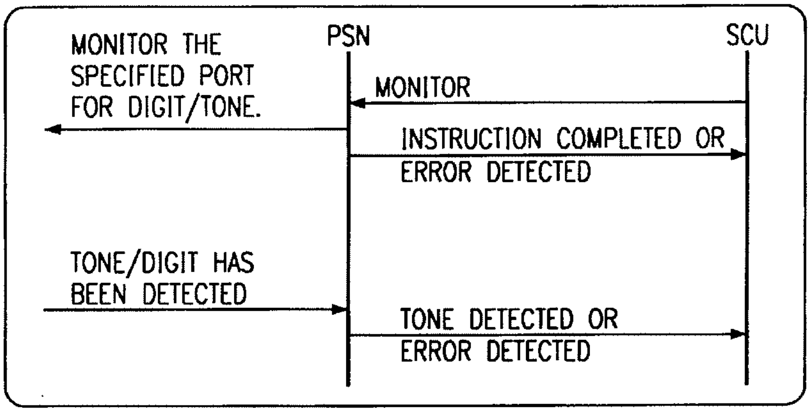


FIG. 19

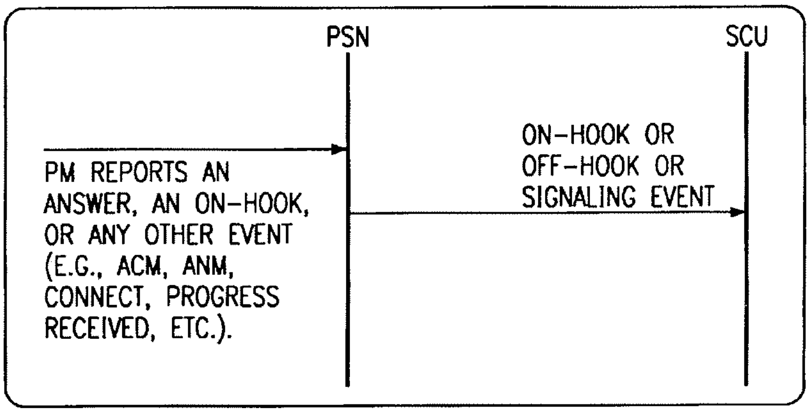


FIG. 20

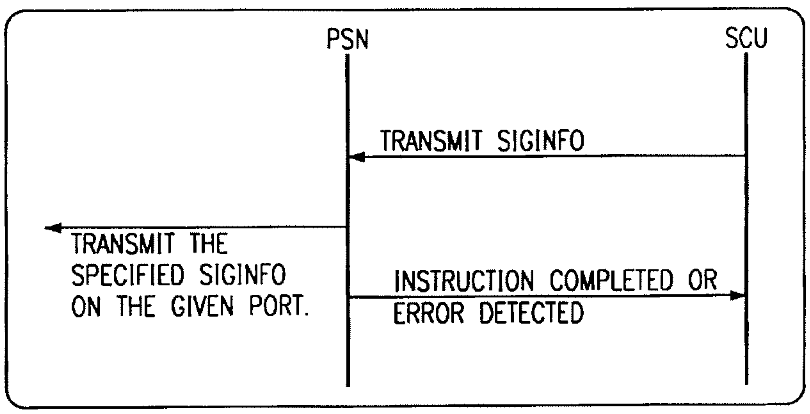


FIG. 21

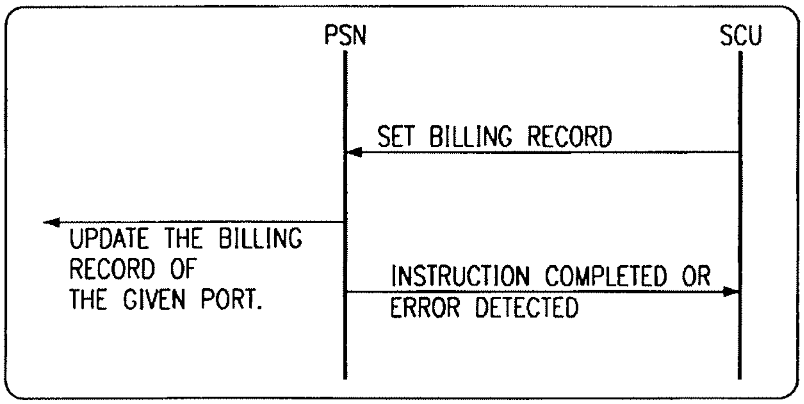


FIG. 22

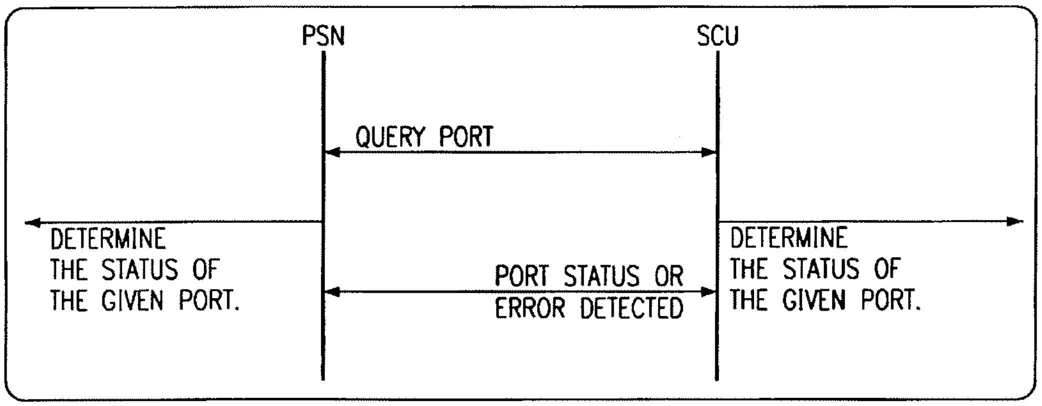


FIG. 23

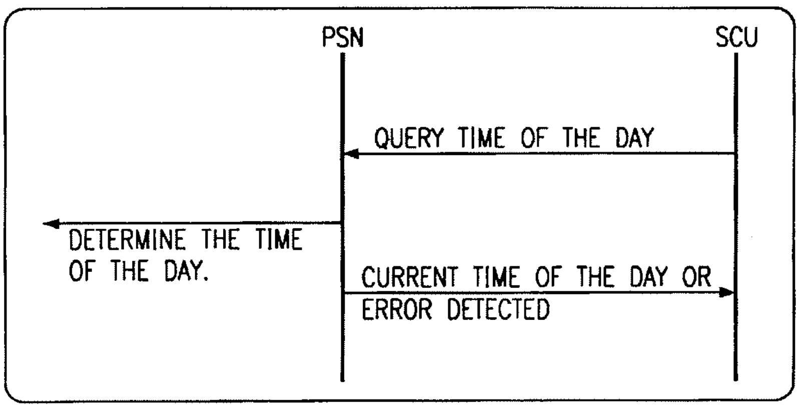


FIG. 24

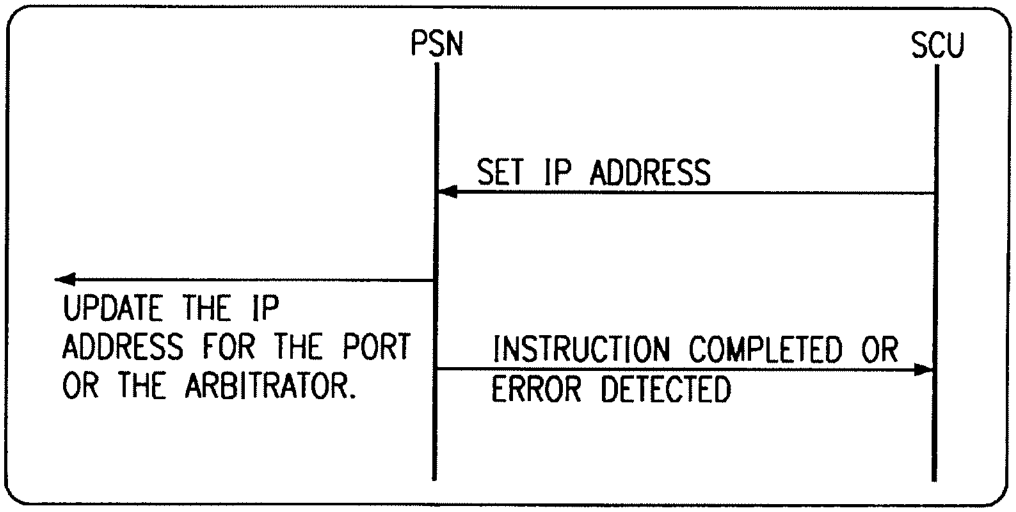

PSN
SCU
SET IP ADDRESS
UPDATE THE IP ADDRESS FOR THE PORT OR THE ARBITRATOR.
INSTRUCTION COMPLETED OR ERROR DETECTED

FIG. 25

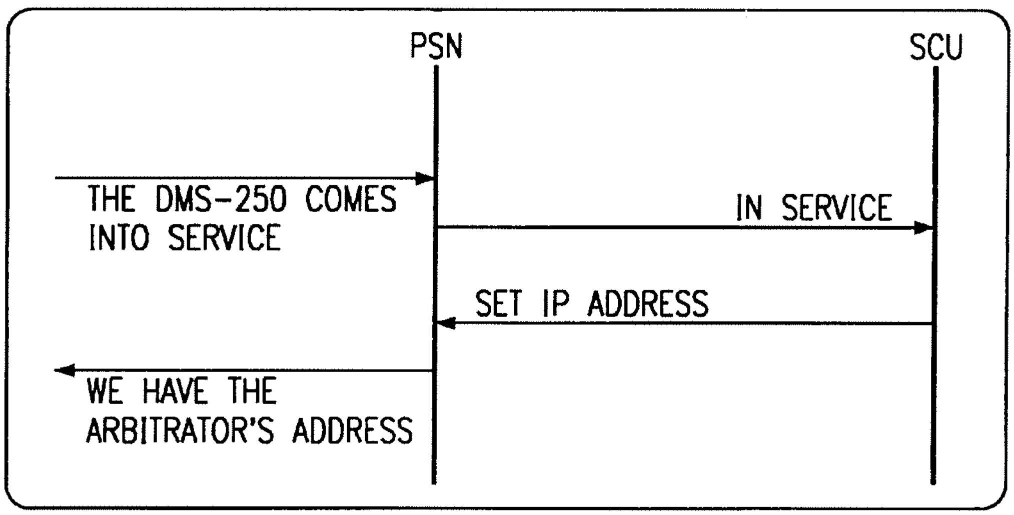

PSN
SCU
THE DMS-250 COMES INTO SERVICE
IN SERVICE
SET IP ADDRESS
WE HAVE THE ARBITRATOR'S ADDRESS

FIG. 26

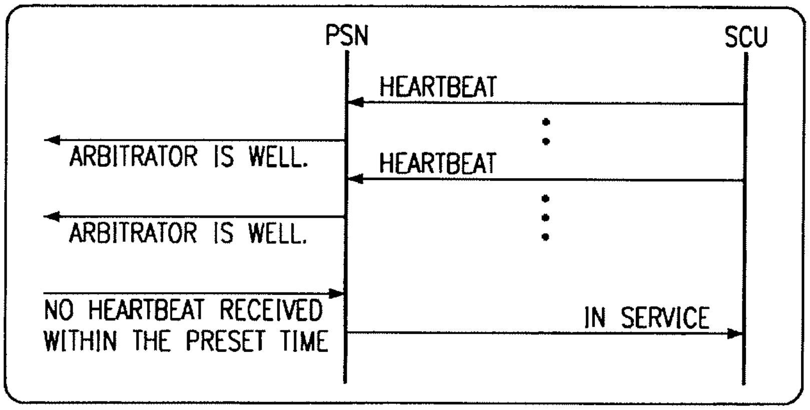

PSN
SCU
HEARTBEAT
ARBITRATOR IS WELL.
HEARTBEAT
ARBITRATOR IS WELL.
NO HEARTBEAT RECEIVED
WITHIN THE PRESET TIME
IN SERVICE

FIG. 27

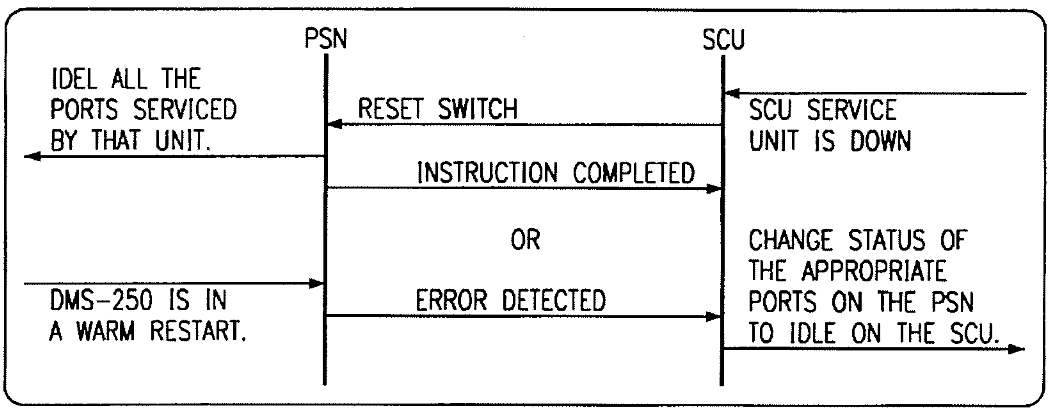

PSN
SCU
IDEL ALL THE
PORTS SERVICED
BY THAT UNIT.
RESET SWITCH
SCU SERVICE
UNIT IS DOWN
INSTRUCTION COMPLETED
OR
CHANGE STATUS OF
THE APPROPRIATE
PORTS ON THE PSN
TO IDLE ON THE SCU.
DMS-250 IS IN
A WARM RESTART.
ERROR DETECTED

FIG. 28

ACTION	SCU ->> SWITCH MATRIX	SWITCH MATRIX ->>SCU	STATE TRANSITIONS FOR PARTY A	STATE TRANSITIONS FOR PARTY B
THE CALL TRIGGERS. THE SCU IS NOTIFIED OF A NEW CALL EVENT.	NEW CALL ACCEPTED (A)	NEW CALL	IDLE HELD → ANSWERED HELD	
SCU ASKS SWITCH MATRIX TO COLLECT DIGITS ON PARTY A	PLAY PPCD COLLECT DIGITS AND REPORT (A)	INSTRUCTION COMPLETED DIGITS COLLECTED	ANSWERED HELD REMAIN IN SAME STATE	
SCU ASKS SWITCH MATRIX TO CONNECT PARTY A TO PARTY B	CONNECT (A,B)	ROUTE SELECTED	ANSWERED HELD → ANSWERED LINKED	IDLE HELD → SEIZED LINKED
SCU ASKS SWITCH MATRIX TO HOLD PARTY A	HOLD (A)	INSTRUCTION COMPLETED	ANSWERED LINKED → ANSWERED HELD	SEIZED LINKED → SEIZED HELD

–CONTINUED–

FIG. 29a

ACTION	SCU->> SWITCH MATRIX	SWITCH MATRIX ->>SCU	STATE TRANSITIONS FOR PARTY A	STATE TRANSITIONS FOR PARTY B
PARTY B ANSWERS OR GOES OFF-HOOK.		OFFHOOK (B)	ANSWERED HELD REMAIN IN SAME STATE	SEIZED HELD → ANSWERED HELD
SCU ASKS SWITCH MATRIX TO RECONNECT PARTIES A AND B	RECONNECT (A,B)	INSTRUCTION COMPLETED	ANSWERED HELD → ANSWERED LINKED	ANSWERED HELD → ANSWERED LINKED
SCU ASKS SWITCH MATRIX TO MUTE PARTY A	MUTE	INSTRUCTION COMPLETED	ANSWERED LINKED → ANSWERED LISTENING	ANSWERED LINKED REMAIN IN SAME STATE
SCU ASKS SWITCH MATRIX TO UNMUTE PARTY A	MUTE	INSTRUCTION COMPLETED	ANSWERED LISTENING → ANSWERED LINKED	ANSWERED LINKED

-CONTINUED-

FIG. 29b

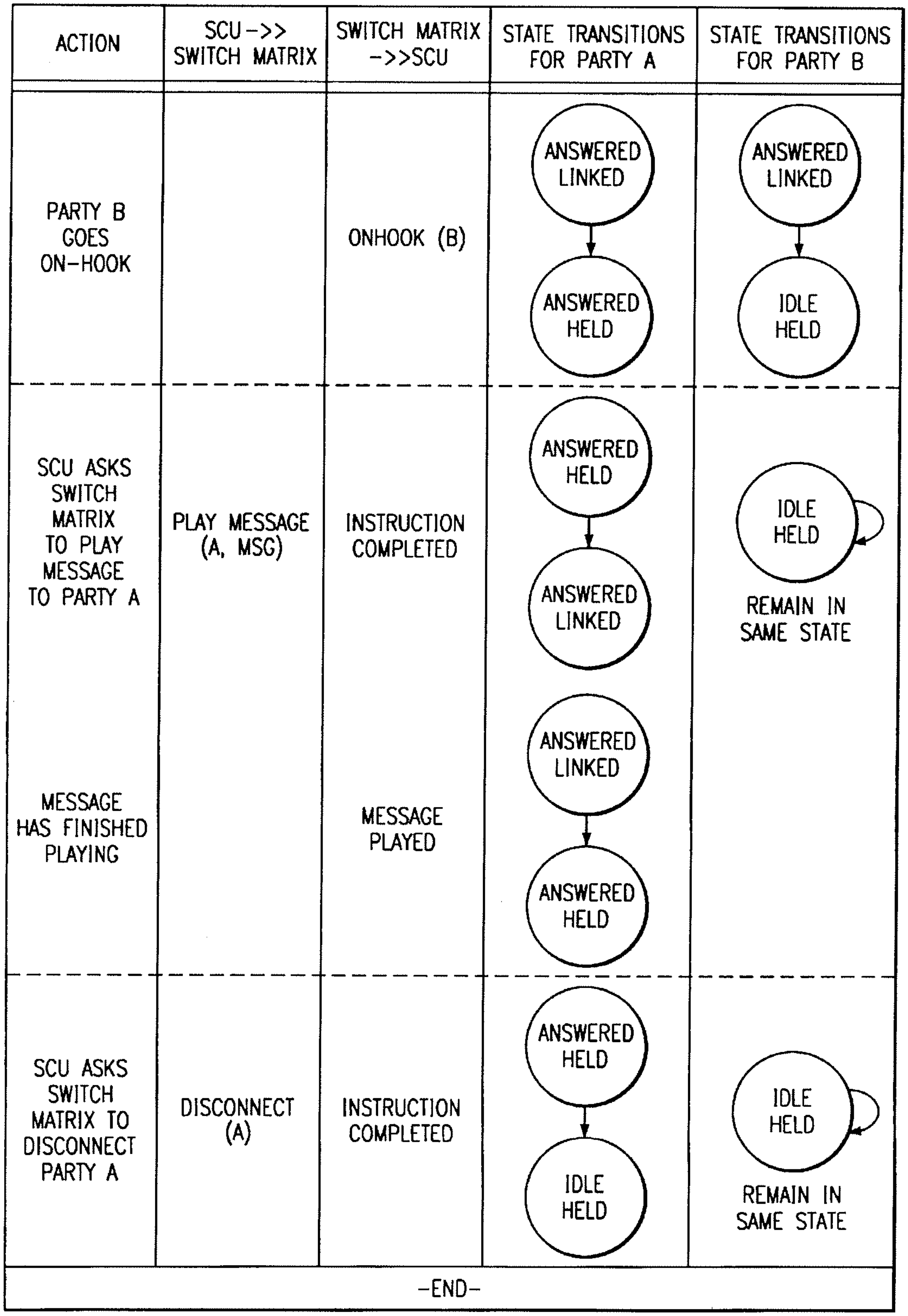

ACTION
SCU ->> SWITCH MATRIX
SWITCH MATRIX ->>SCU
STATE TRANSITIONS FOR PARTY A
STATE TRANSITIONS FOR PARTY B
PARTY B GOES ON-HOOK
ONHOOK (B)
ANSWERED LINKED
ANSWERED HELD
ANSWERED LINKED
IDLE HELD
SCU ASKS SWITCH MATRIX TO PLAY MESSAGE TO PARTY A
PLAY MESSAGE (A, MSG)
INSTRUCTION COMPLETED
ANSWERED HELD
ANSWERED LINKED
IDLE HELD
REMAIN IN SAME STATE
MESSAGE HAS FINISHED PLAYING
MESSAGE PLAYED
ANSWERED LINKED
ANSWERED HELD
SCU ASKS SWITCH MATRIX TO DISCONNECT PARTY A
DISCONNECT (A)
INSTRUCTION COMPLETED
ANSWERED HELD
IDLE HELD
IDLE HELD
REMAIN IN SAME STATE
-END-

FIG. 29c

PROGRAMMABLE SERVICE ARCHITECTURE FOR CALL CONTROL PROCESSING

CLAIM OF PRIORITY

This application claims priority from U.S. Provisional Patent Application No. 60/020,713 of Kent W. Smith et al., entitled "PROGRAMMABLE SERVICE ARCHITECTURE" filed Jun. 13, 1996.

CROSS-REFERENCE TO RELATED APPLICATIONS

The following co-pending United States Patent Applications filed concurrently with this application contain related information and are incorporated herein by reference:

U.S. patent application Ser. No. 08/865,887 of Geetha R. Ram et. al. filed may 30, 1997; and

U.S. patent application Ser. No. 08/865,692 of Geetha R. Ram et. al. filed May 30, 1997.

TECHNICAL FIELD OF THE INVENTION

The present invention relates to an apparatus and method for call control processing in a telecommunications network and, in particular, to an apparatus and method of controlling the call processing of enhanced service calls.

BACKGROUND OF THE INVENTION

In the telecommunications network's classic structure, intelligence resides in the internals of the individual switching systems within the network. In this structure when a telecommunications operating company needs to evolve their network, keep up with standards or differentiate their products, they usually contract with the vendor of the switching equipment for new switch-based development. While this is excellent business for the switching equipment vendor, it requires the operating company to be locked into the particular vendor's delivery schedule and quality of output. With the complexity of switch-based software continually growing, updates and changes to the software required to implement a new service (or modify existing services) are taking longer to complete, and often degrades the quality of the processing. In the rapidly changing telecommunication marketplace success has a lot to do with introducing services quickly and with exceptional quality. Two existing enhancements to the classic telecommunications network help give the operating company independence from the switch vendors and improve service deployment. These include (1) Intelligent Networking (IN) or Advanced Intelligent Networking (AIN) and (2) programmable switching matrices.

Intelligent Networking is driven by standards that have defined (and include) components such as Service Control Points (SCP) and Service Switching Points (SSP) communicating via a common channel signaling system 7 (CCS7) infrastructure. The IN components and standards decouple service development from internal switch implementation and allows an operating company to develop services in isolation of the switch vendors' development and deployment schedules. The benefits of these standards and the resulting changes to the network architecture are that services can be written independently from the switch, implemented by third-parties or the operating company themselves, and work with switches manufactured by different vendors. The disadvantages of IN, however, are that the standards have been slow to evolve and the complexity is high thus requiring a phased implementation from switch vendors, and the functionality defined by the CCS7 messaging (e.g., TCAP) leaves little room for differentiation of service offerings.

Taking advantage of the slowness in which the IN standards have been defined, implemented and deployed in the switching network, some telephony equipment vendors have implemented stand-alone programmable switching matrices. These products have been used to prototype and deploy services into the public switch network. Examples of such services include voice dialing, televoting, and debit card services. These stand-alone systems are connected as a terminal appliance, typically via integrated services data network (ISDN), primary rate interfacing (PRI) or T1 connections, onto the existing backbone network as service overlay networks.

The advantages of stand-alone programmable switching matrices are the same as with the IN but there are several disadvantages. The standalone programmable switching matrices do not offer the reliability or the capacity required for optimal network implementation. Additionally, the different computing platforms added into the network to support these overlay systems introduce maintenance and network management complexities resulting in high operations and sustaining costs. Lastly, with these stand-alone overlay programmable switching matrices there is an inefficient use of resources as both systems are tied up for the duration of the service.

SUMMARY OF THE INVENTION

In accordance with the present invention, there is provided an apparatus for controlling and processing a service call in a telecommunications system. In one embodiment of the present invention, the apparatus includes a programmable switch matrix having at least two ports, a first port and a second port. The programmable switch matrix also includes one or more predetermined triggers for detecting when processing of a service call associated with the first port is desired to be controlled externally by a service control unit. A service node outputs an event notification message to the service control unit in response to the detection of the service call and receives one or more primitives (instructions) from the service control unit. Call processing circuitry within the programmable switch matrix connects the first port to the second port in response to the one or more primitives received by the programmable service node from the service control unit. The service control unit is located external to the programmable switch matrix and controls the programmable switch matrix and processing of the service call through a communications link between the programmable switch matrix and the service control unit.

In another embodiment of the present invention, the apparatus includes a service control unit having one or more application software programs for controlling call processing of the service call. A programmable switch matrix is provided having a plurality of ports, and at least a first port and a second port. The programmable switch matrix generates service control unit service call request when call control processing of the service call received on the first port is desired to be controlled by the one or more of the application software programs within the service control unit. An event notification message is generated and output by the programmable switch matrix to the service control unit in response to the service call request. The event notification message includes call information associated with the service call. One or more primitives (instructions) are received from the service control unit and the programmable switch matrix connects the first port to the second port in response to the one or more received primitives. The service control unit is located external to the programmable switch matrix and externally controls the programmable

switch matrix and processing of the service call. A first data link provides a communication path between the programmable switch matrix and the service control unit for carrying the one or more primitives from the service control unit to the programmable switch matrix and for carrying the event notification message from the service control unit to the programmable switch matrix. The apparatus also includes a media resource unit for generating and outputting a message to at least one of the plurality of ports. A second data link between the media resource unit and the programmable switch matrix carries the message to the at least one of the plurality of ports and a third data link between the service control unit and the media resource unit is used for controlling the output of the message from the media resource unit.

In accordance with the present invention, there is provided a method of controlling and processing a service call in a telecommunications system. In another embodiment of the present invention, a service call is received on a first port of a switch matrix and it is determined from call information associated with the received call that processing of the received call should be controlled by a predetermined service application program. A new call event notification message is transmitted to a service control unit notifying the service control unit that the received call should be controlled by the service application program. The service control unit located externally to the switch matrix and includes the service application program. The service application program controls the processing of the received call. One or more primitives are transmitted from the service control unit to the switch matrix instructing the switch matrix to connect the first port to a second port of the switch matrix.

In yet another embodiment of the present invention, the method includes the steps of receiving a call on a first port of a switch matrix wherein the received call includes call information; performing in-switch call processing of the received call until a trigger is detected; triggering when the call information of the received call meets one or more predetermined trigger criteria within a trigger database; sending a first data message to a service control unit in response to the triggering, the first data message includes a first set of data defining a type of the received call and a second set of data defining an address of the first port. The service control unit is located external to the switch matrix. The method also includes the steps of sending a second data message from the service control unit to the switch matrix to notify the switch matrix that the first data message was received and that processing of the received call will be controlled by the service control unit; selecting a predetermined service application program to handle control processing of the received call in response to the call type data of the first message; executing the selected service application program, the selected service application program generating one or more instructions for performing one or more actions within the switch matrix; sending the one or more instructions to the switch matrix to control processing of the received call; and connecting the first port to a second port of the switch matrix in response to the one or more instructions.

In accordance with the present invention, there is also provided a service programming interface protocol for use between a switch matrix and an external service control unit having one or more service application programs for execution for controlling a service call received on one of a plurality of ports of the switch matrix. The service programming interface protocol includes a plurality of primitives for transmission from the service control unit to the switch matrix for controlling the switch matrix. The primitives include a bridge primitive for instructing the switch matrix to connect a plurality of predetermined ports to each other within the switch matrix, a collect digits and report primitive for instructing the switch matrix to receive a specified number of multifrequency digits at a predetermined port and transmit the received digits to the service control unit, a connect primitive for instructing the switch matrix to connect a predetermined first port of the switch matrix to another port of the switch matrix, a disconnect primitive for instructing the switch matrix to disconnect a predetermined first port of the switch matrix from another predetermined port of the switch matrix, and a new call accepted primitive for instructing the switch matrix that the processing of the service call will be controlled by the service control unit. The interface protocol also includes a plurality of event notification messages for transmission from the switch matrix to the service control unit to notify the service control units of one or more events occurring at the switch matrix. The plurality of event notification messages includes a digits collected event notification message for informing the service control unit that a specified number of digits have been collected from a predetermined port and the identity of the collected digits, an instruction completed event notification message for informing the service control unit that the switch matrix has completed one or more actions in response to one or more primitives previously received from service control unit, a new call event notification message for informing the service control unit that a triggering event has occurred for the received call and that the service call is to be controlled by the service control unit, an off-hook event notification message for informing the service control unit that a predetermined port is off-hook, and an on-hook event notification message for informing the service control unit that a predetermined port is on-hook.

In still yet another embodiment of the present invention, there is provided a method of communicating, via a communications link, between a switch matrix and an external service control unit having one or more service application programs for controlling a service call received on one of a plurality of ports of the switch matrix, The method includes the steps of instructing, through one or more primitives sent from the external service control unit to the switch matrix, the switch matrix to (1) connect a plurality of predetermined ports to each other within the switch matrix, (2) receive a specified number of multifrequency digits at a predetermined port and transmit the received digits to the service control unit, (3) connect a predetermined first port of the switch matrix to another port of the switch matrix, (4) disconnect a predetermined first port of the switch matrix from another predetermined port of the switch matrix, and further to instruct the switch matrix that the processing of the service call will be controlled by the service control unit. The method further includes the steps of informing, through one or more event notification messages sent from the switch matrix to the external service control unit, the service control unit that (1) a specified number of digits have been collected from a predetermined port and the identity of the collected digits, (2) the switch matrix has completed one or more actions in response to one or more primitives previously received from service control unit, (3) a triggering event has occurred for the received call and that the service call is to be controlled by the service control unit, (4) a predetermined port is off-hook, and (5) that a predetermined port is on-hook.

The foregoing has outlined rather broadly the features and technical advantages of the present invention in order that the detailed description of the invention that follows may be better understood. Additional features and advantages of the invention will be described hereinafter which form the subject of the claims of the invention. It should be appreciated by those skilled in the art that the conception and the specific embodiment disclosed may be readily utilized as a basis for modifying or designing other structures for carry-

ing out the same purposes of the present invention. It should also be realized by those skilled in the art that such equivalent constructions do not depart from the spirit and scope of the invention as set forth in the appended claims.

BRIEF DESCRIPTION OF THE DRAWING

For a more complete understanding of the present invention, and the advantages thereof, reference is made to the following description taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a schematic block diagram of the present invention that includes a programmable switch matrix and a service control unit;

FIG. **2** is a more detailed schematic block diagram of the programmable switch matrix illustrated in FIG. **1**;

FIG. **3** is a simplified block diagram of the present invention;

FIG. **4** is a functional block diagram of the programmable switch matrix;

FIG. **5** is a functional block diagram of the software architecture of a programmable service node of the present invention;

FIG. **6** is a schematic block diagram of the service control unit;

FIG. **7** is a functional block diagram of the software architecture of the service control unit of the present invention;

FIG. **8** illustrates the software architecture and message flow within the call distribution manager shown in FIG. **7**;

FIG. **9** illustrates a functional flow of a request for data from the service application to a database;

FIG. **10** illustrates packaging of an instruction message (primitive);

FIG. **11** illustrates packaging of an event notification;

FIGS. **12***a*, **12***b* and **12***c* illustrate a finite state machine which manifests operation of the programmable service node;

FIG. **13** illustrates a message flow for a New Call event notification;

FIG. **14** illustrates a message flow for a Connect primitive;

FIG. **15** illustrates a message flow for a Play Message primitive;

FIG. **16** illustrates a message flow for Collect Digits & Report, and Play Prompt, Collect Digits & Report primitives;

FIG. **17** illustrates a message flow for Disconnect, Reconnect, Hold, Mute And Stop Message primitives;

FIG. **18** illustrates a message flow for a Bridge primitive;

FIG. **19** illustrates a message flow for a Monitor primitive;

FIG. **20** illustrates a message flow for a signaling event notification, including an on-hook and off-hook;

FIG. **21** illustrates a message flow for a Transmit SigInfo primitive;

FIG. **22** illustrates a message flow for a Set Billing Record primative;

FIG. **23** illustrates a message flow for a Query Port primitive;

FIG. **24** illustrates a message flow for a Query Time primitive;

FIG. **25** illustrates a message flow for Set IP Address primitive;

FIG. **26** illustrates a message flow for an In Service event notification;

FIG. **27** illustrates a message flow for a Heartbeat primitive;

FIG. **28** illustrates a message flow for a Reset Switch primitive; and

FIGS. **29***a*, **29***b*, and **29***c* illustrate a call state and transition example for an enhanced service call.

DETAILED DESCRIPTION OF THE INVENTION

With reference to the drawings, like reference characters designate like or similar elements throughout the drawings.

A. Distributed Programmable Service Architecture

Now referring to FIG. **1**, there is illustrated, in accordance with the present invention, a distributed programmable service architecture **10** that includes a programmable switch matrix **24** embodied within a public switched telephone network (PSTN) **20**. Connected to the PSTN **20** are multiple network user appliance terminals **22**. As will be appreciated, user terminals **22** may be telephone sets (Plain Old Telephone Sets—POTS), mobile base stations (MBS), integrated services data networks (ISDN), private branch exchanges (PBX), and any other user terminals interconnected to the PSTN **20**.

The PSTN **20** is representative of a telecommunications network infrastructure having a plurality of switching systems and other network elements which are interconnected by transmission facilities, whereby user terminals **22** (and users thereof) are able to communicate. The programmable switch matrix **24** includes a programmable service node (PSN) **28** that provides external programmability to the programmable switch matrix **24**. The programmable switch matrix **24** is communicatively coupled via a communications link **40** to a service control unit (SCU) **34** within a service control platform **32**. In addition, the programmable switch matrix **24** is communicatively coupled via a communications link **42** to a media resource unit (MRU) **36**, respectively, within a media resource platform **33**.

The programmable service architecture **10** characterizes a client-server type arrangement whereby enhanced telephony services may be deployed into the PSTN **20**. The programmable switch matrix **24** functions as a conventional switching system that performs traditional call processing as an autonomous in-switch operation. The programmable service node (PSN) **28** provides a server mode operation in which call processing and hardware resources of the programmable switch matrix **24** (client) are controlled by the external service control platform **32**, specifically the SCU **34** thereof. The programmable switch matrix **24** is preferably deployed in a tandem/toll office within the PSTN **20** which positioning allows it to be accessed by a larger subscriber base and billing of charges for service utilization may be readily effected with existing network capabilities.

The service control platform **32** provides functionality for implementing telephony services, wherein software applications (service software programs) defining such services are executed by the SCU **34** that control the call processing of a desired service call. The PSN **28** communicates with the SCU **34** over the communications link **40** utilizing a service programming interface signaling protocol. In the preferred physical embodiment, the communications link **40** includes an Ethernet link, but may include any other communications standards and hardware such as IEEE 802.3 (10BaseT), Fiber Distributed Data Interface (FDDI), Asynchronous Transfer Mode (ATM), and the like.

The media resource unit (MRU) **36** provides additional call processing capabilities and functions (enhanced services) which may not have been directly provisioned in the programmable switch matrix **24**, including for example, voice messages or announcements, tone, interactive voice response, advanced and/or flexible voice recognition, voice record and store, fax server, and the like, that are provided to the PSTN **20**. The MRU **36** interacts with calls connected

via the communications link **42** to the programmable switch matrix **24** under the control of the SCU **34**. The communications link **42** includes capabilities for voice/data communications, and may include DS1 digital trunks, or any other appropriate communications system. The MRU **36** and the SCU **34** communicate via a communications link **38** that may be similar to the communications link **40**.

An optional service control point (SCP) **26** and/or AIN database **27** may be included within the programmable service architecture **10**. The SCP may contain the AIN database **27**, or the AIN database **27** may be physically separate. A communications link **30** is provided between the programmable switch matrix **24** and the SCP **26** and/or AIN database **27**.

The AIN database **27** generally contains translation numbers (for a given trigger criteria) and/or routing information for a particular type of call that is queried when the programmable switch matrix **24** receives and triggers on a service call that requires (or desires) AIN service. Typically, AIN services include 1-800 translations, etc., and the AIN database **27** provides translation and routing information (i.e., provides the real number to switch matrix **24** for routing).

As will be appreciated, the programmable service architecture **10** may also include one or more external informational databases **56** interconnected between the SCU **34** via a communications link **58** and between the programmable switch matrix **24** via a communications link **60**, respectively, for additional information capabilities if desired.

B. Programmable Switch Matrix and Service Node

1. Hardware

Now referring to FIG. **2**, there is shown a block diagram of the programmable switch matrix **24** in accordance with the present invention. Functionally, the programmable switch matrix **24** is divided into three layers: service processing **72**, signaling and connectivity **74**, and physical access **76**. Although every layer includes many central processing units (CPUs), the service processing layer **72** as the top layer contains the processing for central call handling, system control, and system management. It also provides the higher-level functions associated with the wide variety of telecommunications services provided by the programmable switch matrix **24**. The signaling and connectivity layer **74** contains the system's major internal messaging component and the call switching fabric. The physical access layer **76** provides terminations, signal processing, service-specific protocol handling, and multiplexing. It also provides interfaces and processors having access to external signaling and data networks, and allows lines and trunks carrying voice or data traffic to connect to the programmable switch matrix **24**.

The physical access layer **76** includes a plurality of input/output lines or trunks **78** for receiving/transmitting voice or data. As will be appreciated, the trunks **78** are characterized by one or more physical connection lines for input/output. Generally, each trunk includes a plurality of ports (or channels, not shown) **79** if the trunk **78** carries time division multiplexed (TDM) signals (if there is no such multiplexing, each trunk may be fairly characterized as a port itself). As will be appreciated, the trunks **78** may carry types of data signals other than TDM, such as frequency division multiplexed (FDM) signals, and the like, or a combination thereof.

The main components of the service processing layer **72** include a core **80** and a plurality of file processors **82**. The core **80** is the main computing resource and provides overall system control, conventional switched call routing, maintenance, and management functions. The core **80** includes duplicated, high capacity, single-chip microprocessors, with duplicated memory, communication buses and interfaces (not shown) to a bus **84**. The file processors **82** enhance the programmable switch matrix's data management capabilities by providing secondary storage and archive functions and allowing both file and database access to their stored data. The file processor **82** typically includes three memory cards, a Small Computer Serial Interface (SCSI) card, and fiber-optic links to the bus **84**, and generally support redundant hard disk drives and digital tape drives (not shown).

The main components of the signaling and connectivity layer **74** include the bus **84** and an enhanced network **86**. The bus **84** is a high speed transaction bus that provides message transport functions for the distributed processors of the programmable switch matrix **24** on a peer-to-peer basis. The bus **84** consists of two load sharing planes, each plane having a CPU, memory, and a set of interfaces (not shown) to the core **80**, the file processors **82**, the enhanced network **86**, etc. The enhanced network **86** handles call switching. It is a time division multiplexing (TDM) switch network that establishes connections between any of the trunks **78** or ports (voice and/or data channels) terminating on the programmable switch matrix **24**. The enhanced network **86** is a non-blocking, single-stage switch that also provides connections for control messages exchanged between a plurality of peripheral processors **88** and the core **80**.

The physical access layer **76** includes the peripheral processors **88**, as well as one or more link peripheral processors **90** and one or more input/output controllers **92**. The peripheral processors **88** provide an interface between the enhanced network (i.e., switching fabric) **86** in the signaling and connectivity layer **74** and the external environment of the programmable switch matrix **24** which, as a tandem office, typically is through digital trunks (providing anywhere from 1 to over 100,000 ports or channels depending on the size). The peripheral processors **88** are controlled by the core **80**, and support it by providing some of the processing required for call handling, as well as the trunk interfaces. For example, once the core **80** has established digital connections among peripheral processors **88**, they can pass voice and data traffic, as well as signaling and control messages among themselves without supervision from the core **80**. The link peripheral processors **90** terminate a number of link types and implement a variety of protocols to connect the programmable switch matrix **24** to the external operations and signaling networks (via communications links **30**, **40**, **60**, and other links between the programmable switch matrix **24** and other components or systems within the PSTN **20**). Examples of these networks include CCS7 signaling links and Ethernet (IEEE 802.3) with TCP/IP and X.25/X.75 for packet switched data. The input/output controller **92** provides the interface between the bus **84** and a variety of serial devices (not shown) including video display terminals, magnetic tape drives, disk drives, data units, modems and printers. Preferably, the programmable switch matrix **24** is equipped with a plurality of six port (or more) conference circuits which are advantageous to support bridging to the MRU **36**.

As will be appreciated, the programmable switch matrix **24** may be based on a DMS-250 programmable switch matrix designed and manufactured by Nortel, Inc.

As illustrated within FIG. **1**, included within the programmable switch matrix **24** is the programmable service node (PSN) **28**, sometimes referred to as the PSN switching system. The programmable service node is implemented by appropriate PSN application software embedded in the core **80**. Execution or operation of the PSN application software thereby provides the functionality of the PSN **28**. The PSN application software includes a finite state machine embodied in software which will be described in detail below.

Now referring back to FIG. **2**, the peripheral processors **88**, in accordance with the present invention, include one or more digital trunk controllers (DTC) **94** that provide trunk

connections (and control thereof) to other machines or systems within or without the PSTN **20**. The DTCs **94** provide trunking that may include per-trunk signalling (PTS) trunks, inter-machine trunks (IMT), and/or signalling system #7 (SS7) trunks, including TX (e.g. T**0**, T**1**, T**2**, etc.) trunks. In addition, one or more integrated services data network (ISDN) digital trunk controllers (DTCI) **96** are provided for ISDN trunking that may include primary rate interfacing (PRI) and/or basic rate interfacing (BRI). A DTC **94** or DTCI **96** connects the programmable switch matrix **24** to digital interoffice carriers and/or private branch exchanges (PBX), the MRU **36**, etc. The DTCs **94** and DTCIs **96** include signal processing resources (not shown), for example, Universal Tone Receivers (UTRs) and Specialized Tone Receivers (STRs) that provide tone generation, tone reception, and digit collection. Another switch resource within the programmable switch matrix **24** is an announcement machine, such as Nortel's Enhanced Digital Recorded Announcement Machine (EDRAM) (not shown). This provides short announcements or messages that are played to one or more of the ports. The MRU **36** typically provides access to longer announcements and messages for bridging to one or more ports. The peripheral processors **88** also include one or more line trunk controllers (LTC) **98** that provide line connections to individual lines (such as user terminals **22**) and/or trunk connections to smaller, local exchanges.

Now referring back to FIG. **1**, the programmable switch matrix **24** (in conjunction with the programmable service node **28**) connects to the SCU **34** through the communications link **40**. Preferably, the communications link **40** is through an Ethernet link, but may include other defined communications links. The communications link **40** is provided by a pair of Ethernet interface units (EIU) **91** included within the link peripheral processor **90** of the programmable switch matrix **24**. The EIUs **91** are configured in an active/standby arrangement. One link is active and carries live data. The other link is inactive, but dedicated, and configured to operate in the event the active link fails. In one embodiment, the physical media used for the communications link **40** is 10 Mbit/s Ethernet 10baseT. Typically, the EIUs **91**, the communications link **40** and the SCU **34** are configured on a subnet dedicated to the programmable service node **28** application. This allows for isolation between the data messaging created by the programmable service node **28** application and other applications (e.g., billing) operating within the programmable switch matrix **24**. As will be appreciated, the SCU **34** also includes a pair of EIUs for interfacing, or other appropriate hardware depending on the type of communications link **40** utilized.

Now referring to FIG. **3**, there is shown a simplified illustration of the present invention. A service call associated with a specific port (port or trunk member), sometimes referred to as an "agent," that is generated by one of the user terminals **22** is received by the programmable switch matrix **24**. If the service call is a "conventional" call, the programmable switch matrix **24** processes the call in accordance with conventional call processing techniques. If the service call is identified as a call that requires (or desires) specialized call control processing, the call enters what is identified as a "server mode." In the server mode, the call or agent (port or trunk member associated with the call) is controlled by the SCU **34**. Accordingly, the SCU **34** provides instructions to the programmable switch matrix **24** (via the programmable service node **28**) to control the processing of the call or agent. As will be appreciated, other additional ports or trunk members may be associated with a call. These may also be characterized as "agents" even though the additional port(s) or trunk member(s) are not the initiator of the service call request, but are involved in the service call. Instructions received from the SCU **34** intended to control any of the resources within the programmable switch matrix are sent by the core **80** to the enhanced network **86** and the appropriate peripheral processor **88** through the bus **84** using internal messaging (see FIG. **2**).

The service communications link **40** provides a mechanism for the transport of data messages to and from the programmable switch matrix **24** and the SCU **34**. When a call or agent is entering the service mode, and while in this mode, the programmable switch matrix **24** and the SCU **34** communicate with one another through messages called primitives, macros, and event notifications, defined below.

A primitive is transmitted from the SCU **34** to the programmable switch matrix **24**. A primitive is an instruction that controls the programmability of the programmable switch matrix **24** and controls the call or agent processing. A macro is a plurality of primitives sent in one message. An event notification provides status information regarding a port (or trunk member), i.e. agent, involved in a service call in response to primitives and/or macros, or in response to a peripheral event on the port. An event notification is transmitted from the programmable switch matrix **24** to the SCU **34**. Primitives and event notifications are described in more detail later.

All required (or desired) service calls are handled according to the agent. No actions are performed on the service call as a whole. Instead, the primitives and event notifications relate to a specific agent, and all actions are performed on the designated agent. This architecture provides flexibility and does not rely on any assumptions about the call. For conventional in-switch call processing (a non-service call), call supervision and processing is based on the originating and terminating agents. In the programmable service node application, there is no concept of an originator or terminator. Each agent is independent of one another and a group of agents may be a part of a service being driven by the SCU **34**. The present invention allows the SCU **34** to control individual agents and mix and match agents as a service call progresses.

Once in the server mode, the conventional call processing architecture is bypassed for that particular call. Accordingly, a master-slave relationship is created wherein the SCU **34** is the master and the programmable switch matrix **24** is the slave performing actions in response to instructions from the SCU **34**.

The programmable switch architecture **10** allows and supports co-existence with AIN servicing and service calls associated with AIN triggering and the AIN database **27**. A service call may trigger on an AIN trigger or an SCU trigger. As will be appreciated, the AIN triggers and SCU triggers may be provisioned within the programmable switch matrix **24**. If a service call invokes an SCU trigger, the SCU **34** is notified and thereafter controls processing of the call. If the service call invokes an AIN trigger, the AIN database **27** (and/or SCP **26**) provides a translation number (and/or simple routing information) to the switch matrix **24** for routing the call to the appropriate trunk and port (to eventually connect with the real number). It will be understood that the AIN database may determine via a link **29** that the service call should be controlled by the SCU **34**. This allows the capability of updating the AIN database **27** with information directing the switch matrix **24** to have the SCU **34** control the service call instead of functioning through the AIN servicing. Since AIN servicing is rather simplistic, the present invention allows new and enhanced services to be added (through the use of the SCU **34**) to the typical AIN services. Moreover, the addition of services to the AIN database requires switch software modifications within the programmable switch matrix **24**. The present invention does not require such switch software modification to add new services.

As will be appreciated, the present invention supports flexible network configurations that may include one or

more programmable switch matrices **24** and/or SCUs **34**. Examples of such configurations include, single switch matrix—single SCU, single switch matrix—multiple SCUs (i.e., the programmable switch matrix is controlled by services executing on multiple SCUs), multiple switch matrices—single SCU, and multiple switch matrices—multiple SCUs.

Now referring to FIG. **4**, there is shown a functional block diagram of the programmable switch matrix **24**, including the programmable service node (PSN) **28**. As described earlier, the PSN **28** comprises the PSN application software (typically executing within the core **80**). Included in the programmable switch matrix **24** is conventional in-switch call processing circuitry (and may include software) **100** and one or more (or database of) triggers **102** (typically configured in the software) for detecting or identifying when a particular service call requires (or desires) call control processing from the SCU **34**. If no such triggering has been detected or identified, the service call proceeds through the conventional call processing circuitry **100**. In general terms, the conventional call processing circuitry **100** provides a switching matrix for routing calls between various trunks **78** and ports etc. (and possibly **42**). If the service call triggers (i.e. meets the triggering criteria), the processing of the call proceeds under the control of the SCU **34** whereby the SCU **34** sends instructions to the programmable switch matrix **24** (via the PSN **28**) for controlling the call processing (by controlling and programming the switching matrix, i.e., the conventional call processing circuitry **100**) of that particular service call (unless the SCU **34** rejects control of the call).

2. Software

Now referring to FIG. **5**, there is illustrated the software architecture of the PSN **28** in accordance with the present invention. A datalink **114** functions to transmit and receive communications (messages, including primitives and event notifications) between the PSN **28** and the SCU **34**. The datalink **114** provides a transport mechanism (i.e., transport layer) for sending/receiving messages to/from the SCU **34** over the communications link **40**. A server **112** is provided which checks the validity of the message from the SCU **34** and sends the message to the appropriate user application (application layer). A client **113** is also provided which receives a message from the user application (application layer) and sends the message to the transport layer (datalink **114**) for transmission to the SCU **34**. In other words, the server **112** receives messages from the transport layer and transfers them to the application layer while the client **113** receives messages from the application layer and transfers them to the transport layer. The transport layer, or Service Programming Interface (SPI) includes the specific primitives and event notifications that are transmitted between the SCU **34** and the PSN **28**, and are described in more detail later.

Within the software architecture of the PSN **28**, there are additional functional elements including a finite state machine (FSM) **110** and a single-ended supervision functional unit **116**. The FSM **110** defines the states, events and transitions for the PSN **28** in a preferred embodiment, and provides the functionality to receive instructions (primitives) from the SCU **34** regarding a particular port (or agent), perform the instructions, and send event notifications to the SCU **34**. The FSM **110** is described in more detail further below. The single-ended supervision **116** provides single-ended supervision (port B answers, port B reports, not port A) of trunks and ports involved in a service call instead of double-ended supervision (port B answers, port A reports). As will be appreciated, in the preferred embodiment, single-ended supervision is utilized, however, double-ended supervision may be used if desired. Single-ended supervision allows call halves (i.e., originating and terminating halves) to be controlled independently of each other.

C. Service Control Unit (SCU)

1. Hardware

Now referring to FIG. **6**, there is shown a block diagram of the SCU **34**. The components of the SCU **34** include an operations controller shelf **200**, one or more service application controller shelves **202**, a local area network (LAN) **204** for interconnecting the operations controller shelf **200** and the service application controller shelf **202** and for providing communications with outside terminals and other peripherals **208** through a communications link **210**. The SCU also includes a pair of interface units **212** (may be EIUs or the like) for providing the communications link **40** to the PSN **28**. As will be appreciated, the LAN **204** may be redundant to ensure that no single failure in the LAN **204** affects operation of the SCU **34**.

The operations controller shelf **200** contains hardware and software resources to provide centralized operations, administration and management (OAM) on all components within the SCU **34**. The operations controller shelf **200** also provides data interfaces for remote access to the OAM system. Processes of the operations controller shelf **200** communicate with all service application controller shelves **202**, and may be connected to the switch OAM system and any operating company's operational support system (OSS). The operations controller shelf **200** includes hardware and software for performing the following functions: monitoring the service application controller shelves **202** to ensure sanity, detecting and correcting faults, handling logs, alarms, controller shelf and card resource management, diagnostics, and operation measurements, providing one common location for the performance of system level administration for all components in the SCU **34**, supporting printing facilities for the SCU **34**, providing modem access capabilities, providing access capability to the user access and security interface, and supporting software installation and upgrades.

Generally, the operations controller shelf **200** includes hardware subsystems (not shown) such as one or more service processors for providing general computing capabilities, one or more input-output processors for I/O processing, and mass storage (including RAM, tapes, disks, and small computer serial interfaces (SCSI)).

The service application controller shelf **202** (or service controller unit) contain the hardware and software resources required to deliver telephony services applications. Each service application controller shelf **202** contains hardware to connect to network switches, MRUs **36**, and other external resources such as voice response units and external databases **56**. The service application controller shelves **202** include hardware subsystems (not shown) including a master service processor for providing general computing capabilities, and data storage. The master service processor is typically a commercially available single-board computer with a UNIX-based operating system (or other desired operating system), and includes a plurality of serial links for ASCII terminal access, etc., a SCSI controller for communication with a disk or DAT drive, and an attachment unit interface (AUI) for connection to the LAN **204**.

Typically, the service application controller shelves **202** are interconnected with one or more MRUs **36** for access to signal and voice processors capable of performing, among other things, speech recognition which includes both speaker-trained and speaker-independent recognition, speech record and playback which provides announcement and message capabilities to support interactive dialogue design, and speech concatenation which allows longer messages to be formed by combining voice prompts with user-recorded information.

The redundant LAN **204** allows the operations controller shelf **200** and service application controller shelf **202** to communicate with each other and with any LAN-connected devices, such as the terminals and other peripherals **208**,

such as terminals gaining access to the OAM system of SCU **34**. Additionally, the LAN **204** may be connected to other processors required for applications such as database systems. The LAN **204** may be connected into a wide area network (WAN), represented by line **214**, allowing remote access to the SCU **34** from any device on the WAN **214**.

The terminals and other peripherals **208** provide a means for users to edit/modify/delete/add service application software programs within the service application controllers **202**.

As will be appreciated, the functionality of the media resource unit (MRU) **36** within the service control platform **32** (see FIG. **1**) may be provided by one or more of the service application controller shelves **202**. Such functionality would be provided by T1 (or DS1, or other appropriate data communications) digital trunks as part of the communications link **42** and including signal and voice processing capabilities within the MRU **36**.

2. Software

Now referring to FIG. **7**, there is illustrated the software architecture **300** of the SCU **34** in accordance with the present invention. This software architecture provides for the development and execution of service applications (i.e. service applications software programs) on the SCU **34** to control call processing within the programmable switch matrix **24**. In general terms, the SCU software **300** functions to discriminate service calls (i.e., identifying the service call, determining which service application should handle the call, and getting the call to the appropriate processor that is executing the particular service application) and process the call (i.e., interacting with the PSN **28** and/or MRU **36**, interacting with external databases if desired, and implementing service logic). The SCU software **300** is grouped into three layers: a data communications layer **302**, a middleware layer **304** and a service layer **306**. Distribution of software into layers allows for easier software maintenance by isolating the functionality.

The data communications layer **302** includes a service programming interface (SPI) protocol handler (SPIPH) **308** for controlling data communications between components of the SCU **34** and the PSN **28**. When a message (i.e., event notification) is received from the PSN **28**, the SPIPH **308** decodes the message and routes it to one or more of the other components (i.e. see FIG. **7**) in the middleware layer **304** depending on whether that particular component is identified as one to receive a particular event notification message (in the preferred embodiment, call-control events are routed to the CCM **316**, non-call-control events to the CDM **314**, and agent-data to the TRK-RM **320**). When a message (i.e., primitive) is sent to the PSN **28** from the SCU **34**, it is encoded by the SPIPH **308** and transmitted to the PSN **28**. Messaging between the SCU **34** and the PSN **28** uses the service programming interface (SPI) and is described in more detail further below.

Similarly, the data communications layer **302** includes an MRU control protocol handler for SCU (MRUCPHS) **310** for controlling data communications between the SCU **34** and the MRU **36**. When a message (i.e., MRU event notification) is received from the MRU **36**, the MRUCPHS **310** decodes the message and routes it to one or more of the other components (i.e. see FIG. **7**) in the middleware layer **304** depending on whether that particular component is identified as one to receive a particular MRU event notification message in the middleware layer **304** (in the preferred embodiment, call-control events are routed to the CCM **316**). When a message is sent to the MRU **36** from the SCU **34**, it is encoded by the MRUCPHS **310** and transmitted to the MRU **36**.

The service layer **306** contains one or more service application software programs **312**. The service layer **306** may also contain a process for simultaneous processing of a plurality of service application software programs **312**, i.e., running more than one process (service application software program) at a time. The service application software programs **312** implement call handling logic for a desired customer service. As will be appreciated, customers can create and modify, as desired, the service application software programs **312** for their own custom applications. This is accomplished by creating the applications software offline and downloading to the SCU **34**. Such applications may include 1-800 call forward (a call made to an 800 number is forwarded to another number in the network), pre-paid debit card calling (a caller may make any number of calls as long as his/her billing has not exceeded some pre-set dollar value), foreign long-distance call-back operations and personal number service (caller dials one number to get to the called party who may be at one of several locations), and similar services and the like.

The middleware layer **304** includes four main components: a call distribution manager (CDM) **314**, a call control manager (CCM) **316**, a database manager (DBM) **318** and a trunk resource manager (TRK-RM) **320**. The middleware layer **304** handles service call discrimination and routing, database queries and responses, as well as some management of databases. As will be appreciated, the data communications layer **302** and the middleware layer **304** may be executed simultaneously (i.e., running multiple instances) on each of the service applications controller units **202** within the SCU. In addition, the service application software programs **312** for each service applications controller unit **202** may also be duplicated (multiple instances), and/or each controller unit **202** may have different service applications programs **312** or may have one or more duplicated on some and not on others.

The call discrimination manager (CDM) **314** initiates service call discrimination and balances distribution across all available processors in the SCU **34**. It also provides first point of contact between the PSN **28** and the SCU **34**. Upon receiving a New Call event notification from the PSN **28**, the CDM **314** determines the service application to control processing of the call based on call criteria. Upon selecting the service application, the CDM **314** allocates or directs the call to the appropriate service application software program **312** that will control the call and perform the service. If there exist multiple instances of the same service applications software program **312** spread out across numerous service applications controllers **202**, then the CDM **314** uses an algorithm to fairly distribute, or balance, new calls (needing the same servicing) across the multiple instances.

The CDM **314** is sometimes called an "arbitrator" and is identified by a particular address, its IP address. The arbitrator receives all New Call Event notifications (sent to its IP address) from a particular PSN **28** and processes the new service request and passes the new call to the appropriate service application. The arbitrator then informs the PSN **28** (through a call control primitive) of the IP address (sometimes referred to as port service information or return address) of the service application handling the call (i.e., agent/port). The PSN **28** then sends all subsequent events related to the call to this return address unless the SCU **34** updates the IP address to which events should be directed. In other words, each component within the SCU **34** (i.e., service application, etc.) has its own IP address to which the PSN **28** sends appropriate event notifications.

Now referring to FIG. **8**, there is illustrated a more detailed description of the software architecture and message flow within the CDM **314**. The CDM **314** includes a call routing unit (CRU) **380**, a service application discriminator (SAD) **382**, a switch supervisor (SS) **384**, and a service application environment manager (SAEM) **386**. The CRC **380** routes the New Call to the various components of the CDM **314** before routing the New Call to the appropriate

service application software program **312**. Upon receiving the New Call, the CRC **380** routes it to the SAD **382** to determine which service application software program **312** should process the New Call. The SAD **382** determines the correct service application software program **312** to handle the New Call. The SAD **382** queries one of the databases (via the DBM **312**) for service application information, and this information is inserted in the Application ID and Service ID fields in the New Call event notification, and the New Call is directed back to the CDM **314**.

The SS **384** monitors other CDMs **314** processing on other service applications controller units **202**. Each CDM **314** is typically responsible for controlling one or more PSNs **28** (i.e., processing the PSN's New Calls). If one CDM **314** were to go out of service, one of the other CDMs **314** is dynamically designated to take control over call distribution/redistribution. The SAEM **386** monitors the service application software programs **312**. Several different service applications **312** may be available to process the different types of calls. In addition, their may be multiple instances of the same service application **312** executing within the SCU **34**. The SAEM **386** communicates with the service applications **312**, thereby providing information to the CDM **314** on the provisioning and availability of the service applications **312**.

Now referring back to FIG. **8**, the CCM **316** utilizes the knowledge of all active agents and traffic messages relating to each agent to map event notifications (both PSN **28** and MRU **36**) back to the correct call instance. The CCM **316** handles requests from the service layer **306** (i.e. service applications **312**) to send primitives to the PSN **28** and instructions to the MRU **36**, including updating the appropriate data structures with new information, invoking a primitive builder to gather data to build the primitive data before being sent to the PSN **28**, and gathering data to build the MRU message before being sent to the MRU **36**. The CCM **316** also handles event notifications from the PSN **28**/MRU **36** and updates any call changes and determines which service to send each event received from the SPIPH **308** or MRUCPHS **310**. In addition, the CCM **316** retrieves trunk information from a trunk database and handles any signaling propagation necessary for a service call. The CCM **316** stores information about each agent (port) and each resource on a given service application controller unit **202**, provides information for insertion into the primitives/instructions for the PSN **28** (or MRU **36**) before being sent to the SPIPH **308** (or MRUCPHS **310**), and performs predetermined signaling event propagation and/or signaling event translation (i.e., from SS7 to PTS) to services from knowledge of telephony.

The call control manager (CCM) **316** includes a primitive builder (PB) **322**, a MRU instruction builder (MRU-IB) **324**, and a signaling manager (SM) **326** (not shown). The PB **322** builds primitives for transmission to the PSN **28**. As will be described further below, primitives contain one or more parameters and/or fields containing information needed to instruct the PSN **28** to perform a given action. The PB **322** locates accurate information (from the service application program **312** running in the service layer) to place in any required fields and/or parameters that are valid in the SPI protocol and are accurate for correct processing/control of the service call on the PSN **28**. Similar to the PB **322**, the MRU-IB **324** builds instructions for transmission to the MRU **36**. The MRU-IB **324** locates accurate information (from the service application program **312** running in the service layer) to place in any required fields and/or parameters that are valid in the MRU protocol and are accurate for correct processing/control of the service call with respect to the functioning of the MRU **36**. Likewise, the SM **326** builds SS7 parameters for primitives transmitted to the PSN **28**. The SM **326** locates accurate information (from the service application program **312** running in the service layer) relating to SS7 signaling information parameters for placement in any required fields and/or parameters that are valid in the SS7 TR **394** protocol and are accurate for correct processing/control of the service call on the PSN **28**.

The database manager (DBM) **318** is a single point of contact for all database needs of the SCU **34**. The DBM **318** provides a front-end for a plurality of database servers: a database management (DBM) server **330**, a MRU database (MRU-DB) server **332**, and the trunk resource manager (TRK-RM) **320**. It coordinates the dispatch of database requests (e.g., query, insert, update, delete) and the reception of database responses (e.g., response, query response). As will be appreciated, the SCU **34** may include one or more databases. In addition, external databases may be used in combination with the SCU **34**. Generally, the DBM **318** receives a database request from a requesting client, forwards the database request to the appropriate database (to the server for that database), receives database responses from the respective database, and forwards the database response to the requesting client.

Illustrated in FIG. **8** are a trunk resource database (TRK-DB) **334** associated with the TRK-RM **320**, an MRU database (MRU-DB) **336** associated with the MRU-DB server **332**, and a translations database (XLAT-DB) **338**, a service discrimination database (SD-DB) **340**, one or more customer-specific databases **342** (internal to the SCU **34**), and one or more external databases **344**, all associated with the DBM server **330**. The MRU-DB **336** provides MRU data information while the TRK-DB **334** provides PSN trunk (port) information. The XLAT-DB **338** provides routing data, the SD-DB **340** provides service identification information, the customer-specific databases **342** provide user-specific information for service applications, and the external databases **344** may provide any other appropriate type of information and may be similar to, or may include the external databases **56** (illustrated in FIG. 1). Now referring to FIG. **9**, pointed arrows **1** through **6** are used to illustrate a typical functional flow of a request for (and receipt of) data from the service application **312** to the MRU-DB **336**.

Now referring back to FIG. **7**, the trunk resource manager (TRK-RM) **320** controls and maintains accurate PSN datafill on the SCU **34** and processes queries from the DBM **318** for trunk information within the trunk database (TRK-DB) **334**. The TRK-RM **320** may also be provided with input/output capabilities to allow user viewing and editing of the database information within the TRK-DB **334** through a graphical user interface (not shown). Upon receipt of agent-data event notifications that contain initial or changed status and/or data (trunk information) from the PSN **28**, the TRK-RM **320** updates the TRK-DB **336** with the appropriate information. Typically, the TRK-RM **320** updates its own copy of the TRK-DB **336** and broadcasts the updates to any other TRK-DBs **336** residing within the SCU **34**. When New Call event notifications are sent to the CCM **316**, the CCM **316** immediately sends a query to the DBM **318** to retrieve information about each agent involved in the service call. This results in a query from the TRK-RM **320** to the TRK-DB **336** and subsequent retrieval of information and response back from the TRK-DB **336** to the DBM **318** (and to the CCM **316**).

D. Service Programming Interface (SPI)

The present invention includes a service programming interface (SPI) that provides the message-based signaling communications, or protocol, between the programmable switch matrix **24** and the SCU **34**. The protocol provides both instructions in the form of primitives and macros that effectively enable the SCU **34** to control processing of service calls on the programmable switch matrix **24** and event notifications that are reported to the SCU **34** from the

programmable switch matrix **24**. Primitives and event notifications are made up of one or more parameters and each parameter contains one or more bytes of information. A primitive (or call control command) is sent to the programmable switch matrix **24** from the SCU **34** to control the flow and manipulation of a service call, and it may affect one or more ports involved in the service call. The programmable switch matrix **24** generates and transmits event notification messages to notify the SCU **34** of events on one or more ports that are involved in a service call. The events that occur at the programmable switch matrix **24** are in response to an SCU instruction message (primitive) or as a result of a peripheral event on a call port. Preferably, an event notification reports one and only one port event at a time to the SCU **34**. These instructions and event notifications are packaged into a message, and sent back and forth between the programmable switch matrix **24** and the SCU **34**.

1. Primitives

A primitive is an instruction that is sent from the SCU **34** to the programmable switch matrix **24**. In the present invention, there are two types of primitives: call control primitives and non-call-related primitives. A call control primitive is an instruction that controls call flow and may affect one or more ports in a call.

Now referring to FIG. **10**, there is illustrated a format for a message **400** from the SCU **34** to the programmable switch matrix **24** containing call control instruction information. The message **400** includes a primitives number field **402** and a primitives field **404**. The data within the primitives number field **402** indicates the number of primitives that are packaged in the message **400**. If the number is greater than one than the message **400** is described as a macro. The primitives field **404** is where the one or more primitives, the exact number being specified in field **402**, are positioned within the instruction message **400**. Associated with each primitive is a unique identifier **406** of the particular primitive, and one or more parameters **408** holding appropriate information for that primitive.

In accordance with one embodiment of the present invention, the following describes primitives that the SCU **34** utilizes in instruction messages to control call processing within the programmable switch matrix **24** (via the programmable service node (PSN) **28**). For each primitive, one or more informational parameters are included in the message. The description of each of the following primitives also includes an identification of particular parameters associated with the primitive. Parameters, and their content, are described in more detail further below.

A Bridge primitive instructs the PSN **28** to bridge or conference (i.e connect) specified ports at the PSN **28** and, optionally, a message which may be either announcements or tones that are non-interruptible. This primitive makes use of the conference circuit resource in the PSN **28** to bridge multiple parties that are associated with the specified ports. If a message is also to be bridged to the ports, then a Message Info parameter is included with this primitive. The message to be bridged may already be playing to one of the parties or it may be a new one that is yet to be started. If the message is already playing on one of the ports, then it is bridged to the other ports. If the message has not yet been started, then the message is bridged and subsequently started. Parameters for the Bridge primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the respective ports to be bridged. A minimum of either two ports are required if the Message Information parameter is not sent or one port is required if the Message Information parameter is sent; and

Message Information: Contains the ID of a message which is to bridge to the ports. Required only in the case that this message is also to be bridged to the multiple parties. The message may or may not be already playing when the Bridge primitive is received by the PSN **28**.

A Collect Digits & Report primitive instructs the PSN **28** to collect a specified amount of dual tone multifrequency (DTMF) digits at a given port and report the collected digits to the SCU **34**. If the port is currently involved in a connection, then it is held whereby the voice path between this port and the connected port is cut in both directions to allow for the digits to be collected on this port. Parameters for the Collect Digits & Report primitive includes include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port on which the digits should be collected;

Digit Collection: Identifies the digits to collect;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port that is to collect the digits; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Connect primitive instructs the PSN **28** to connect a given port in the call to an available member (port) of another specified trunk agency. The terminating agent is specified by the external trunk group number. An idle member of the given trunk group is identified and the port is connected to this trunk member. The signalling information (that is contained in the Signalling Information parameter) is outpulsed on the destination trunk member. Also, voice path in both directions is established for this two party call. Parameters for the Connect primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port for connection;

Destination Trunk Group: Identifies the destination trunk;

Billing Information: Contains information to update the billing records of the original port and the destination port. This information may also be provided in one or two Billing Information parameters;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the original port and the destination port. This information may also be provided in one or more Port Service Information parameters;

Bearer Capability: Is used if the SCU **34** overwrites the original port's bearer capability; Signaling Information: Contains information for one of the following: (i) digits to outpulse if the port is a per-trunk signaling (PTS) port; (ii) IAM message parameters if the port is a Signaling System 7 (SS7) ISUP port; or (iii) SETUP message parameters if the port is an ISDN primary rate (PRI) port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Disconnect primitive instructs the PSN **28** to disconnect a specified port from a call in which the port is currently active. Parameters for the Disconnect primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to disconnect;

Billing Information: Contains information to update the billing records of the port;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Signalling Information: Identifies the cause for the disconnection or release. This parameter includes one of the following: (i) REL message parameter if the port is an SS7 ISUP port; or (ii) DISC/REL message parameters if the port is an ISDN PRI port.

An Error Detected primitive is sent by the SCU **34** to the PSN **28**, when the SCU **34** has detected a fatal, non-fatal, or non-service affecting error for a port in the server mode. Parameters for the Error Detected primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port on which the error occurred;

Error Type: Identifies the error is fatal or non-fatal;

Error Cause: Identifies the reason or cause of the error:

Port Status: Indicates the status of the port that is in error;

Instruction ID: Identifies the instruction in error; and

Parameter ID: Identifies the parameter in error.

A Flow Control primitive is sent by the SCU **34** to the PSN **28** to initiate flow control based on specified duration and gap indices. The parameters of the Flow Control primitive include:

Instruction Tag: (described further below);

Flow Control: Contains information indicating the maximum length of time the rate of New Call events may be controlled (duration), and indicating the severity of the control or the minimum length of time between consecutive PSN calls allowed while the control is effective (gap).

A Hold primitive instructs the PSN **28** to put a specified port on hold (i.e., cut the voice path in both directions with the connected party). Parameters for the Hold primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to be put on hold;

Billing Information: Contains information to update the port's billing records; and

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Monitor primitive instructs the PSN **28** to monitor a specified port for given digits and/or tones using the specialized tone receiver (STR) with which the programmable switching matrix **24** is equipped. Parameters for the Monitor primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to monitor;

Monitor Mask: Contains information indicating the tones/digits to monitor. Bits are set to start or cancel monitoring for the appropriate tone/digit;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Mute primitive instructs the PSN **28** to cut or re-establish (mute/non-mute) the voice path between a given port and connected party in the direction of the port to the connected party. For example, in a call from A to B where A and B are talking, Mute (A) cuts the voice path from A to B, but the voice path from B to A is not affected in anyway. This means that A can hear B, but B cannot hear A. If the voice path is already cut in the specified direction, then receiving a Mute primitive will un-mute the connection or re-establish the voice path between the two ports. Parameters for the Mute primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to mute (if currently un-muted) or un-mute (if currently muted);

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A New Call Accepted primitive instructs the PSN **28** that a new call event notification was accepted and the port specified in the new call event notification will be controlled by the SCU **34** with further instructions. Parameters for the New Call Accepted primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port that was accepted;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A New Call Rejected primitive instructs the PSN **28** that a new call event notification was rejected and that the port specified in the new call event notification will not be controlled by the SCU **34**. In this case the port will be routed to a PSN failure treatment. Parameters for the New Call Rejected primitive include:

Port Information: Identifies the port that was rejected; and

Billing Information: Contains information to update the port's billing records.

A Play Message primitive instructs the PSN **28** to connect a specified port to a message which may include either an announcement or a tone. The message being played is generally uninterruptible by the port. Parameters for the Play Message primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to play the message to;

Message Information: Identifies the message to play;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Play Prompt, Collect Digits & Report primitive instructs the PSN **28** to play a message (i.e., an announcement or a tone) and initiate digit collection on a specified port. Generally, the message may be interruptible and as soon as the first digit is dialed, the message stops and the digit collection on the port continues. Parameters for the Play Prompt, Collect Digits & Report primitive include:

Session ID: (described further below):

Instruction Tag: (described further below);

Port Information: Identifies the port to play the message to and collect digits from;

Digit Collection: Indicates the digits to collect;

Message Information: Identifies the message to play;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Reconnect primitive instructs the PSN **28** to reestablish a connection between any two ports in a service call. Parameters for the Reconnect primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the "from" port (port A) in the connection;

Port Information: Identifies the "to" port (port B) in the connection;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Set Billing Record primitive instructs the PSN **28** to update billing records for a specified port with given information. This primitive may be received at any time during the call to update the appropriate billing information for the given port. Parameters for the Set Billing Record primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Stop Message primitive instructs the PSN **28** to stop playing a message that is currently playing to one or more specified ports. If the port is connected only to the message and no other port, then the port is held. If the port is connected to other ports in addition to the message (i.e., was bridged to several other ports and the message), then it remains connected to the other ports and the bridged ports no longer hear the message. Parameters for the Stop Message primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port that the message is played to;

Message Information: Identifies the message to be stopped;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

A Transmit SigInfo primitive instructs the PSN **28** to transmit signaling information on a specified port. Parameters for the Transmit SigInfo primitive include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to transmit signaling information on;

SigInfo Mask: Gives control to the SCU **34** over which of the optional SS7/PRI messages are reported to the SCU **34** in the Signaling Event notification;

Signaling Information: Contains information for one of the following: (i) digits to outpulse if the port is a per-trunk signaling (PTS) port; (ii) any SS7 message if the port is a Signaling System 7 (SS7)ISUP port; or (iii) any PRI message if the port is an ISDN primary rate (PRI) port;

Billing Information: Contains information to update the port's billing records;

Port Service Information: Identifies the return address used to report primitive responses and event notifications to the SCU **34** for the port; and

Control Information: Contains information concerning what actions to perform when primitives have been successfully processed on a given port.

In addition to sending messages having the individual call control primitives described above, the SCU **34** may send multiple primitives in a single message. These primitive instructions collectively are called macros. Following are some examples of macros.

If there are five parties in a service call at any given time, then a "take down call" macro can be implemented. This macro consists of five "Disconnect" primitives, one for each party in the call. Upon execution of this macro, the entire five party call is taken down. Another example would be the "Preempt" macro. Consider a call with parties A and B talking. The goal is for another party C to preempt this call, thus leaving B connected to a message and A now talking to C. This macro can be implemented by sending two primitives, namely a "Reconnect" primitive for A to C followed by a "Play Message" primitive for B.

It will be understood that execution of one primitive may or may not be completed by the PSN **28** before initiating execution of the next. For example, if a macro consists of a "Play Message" primitive followed by a "Connect" primitive, then the message (i.e., announcement) is started on the specified port. Immediately after, a "Connect" primitive is executed that aborts the message being played and connects the port to another destination trunk member.

2. Event Notifications

FIG. **11** illustrates a format for a message **410** from the PSN **28** to the SCU **34** containing event notification information. The message **410** includes an event notification field

412 containing an identifier of the event, and a parameters field 414 containing one or more parameters holding appropriate information for the particular event.

The PSN 28 uses event notification messages to notify the SCU 34 of events on the port(s) that are involved in a service call. The events that occur at the programmable switch matrix 24 may be in response to the SCU 34 primitive/macro instruction or as a result of a peripheral event on a port. An event notification message is used to report a port event to the SCU 34 in response to a primitive. Accordingly, each primitive in a macro will have its own corresponding event notification message reported to the SCU.

In accordance with one embodiment of the present invention, the following describes event notification messages that are sent by the PSN 28 to the SCU 34. For each event notification, one or more informational parameters are included in the message. The description of each of the following event notifications also includes an identification of particular parameters associated with the event notification. Parameters, and their content, are described in more detail further below.

A Digits Collected event notification message is reported to the SCU 34 when (i) the PSN 28 has collected the specified number of digits from the specified port, (ii) an end delimiter has been dialed by the user on the specified port, or (iii) the digit collection has timed out. This event notification is sent in response to the following primitives received from the SCU 34: Collect Digits & Report; and Play Prompt, Collect Digits & Report. Parameters for the Digits Collected event notification include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port on which digits were collected; and

Digits Collected: Indicates the digits that were collected.

An Error Detected event notification message is reported to the SCU 34 when the PSN 28 detects an error. Examples include an error detected while parsing an incoming message from the SCU 34, unavailable resources on the PSN 28 to complete the execution of the received instruction, internal processing and resource errors on the PSN 28, hardware failures, force releases, etc. The Error Detected event notification may be sent in response to any SCU primitive. This notification may also be received from the SCU to report a non-fatal error. Parameters for the Error Detected event notification include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port on which the error occurred;

Error Cause: Identifies the cause for error;

Port Status: Indicates the status of the port that is in error;

Instruction ID: Identifies the instruction in error; and

Parameter ID: Identifies the parameter in error.

An Instruction Completed event notification message is reported to the SCU 34 when the PSN 28 has completed action on a received primitive. This event notification is sent in response to any of the following primitives received from the SCU 34: Bridge, Collect Digits & Report, Disconnect, Hold, Monitor, Mute, Play Message, Play Prompt, Reconnect, Reset Switch, Set Billing Record, Set IP Address, Stop Message, Transmit SigInfo. Parameters for the Instruction Completed event notification include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port on which the specified instruction was executed;

Error Type: Indicates whether a fatal or non-fatal error has occurred;

Port Status: Indicates the status of the port; and

Instruction ID: Identifies the instruction that was executed.

A Message Played event notification message is reported to the SCU 34 when the message on a specified port has ended (i.e., finished playing). This event notification is sent in response to the Play Message primitive received from the SCU 34. Parameters for the Message Played event notification include:

Session ID: (described further below);

Instruction Tag: (described further below); and

Port Information: Identifies the port on which the message was played.

A New Call event notification message is reported to the SCU 34 when an agent (port) is to be controlled by the SCU 34 (by triggering). Parameters for the New Call event notification include:

Switch ID: Identifies the programmable switch matrix;

Port Information: Identifies the port(s) in a call when it is determined that the call is to be controlled by the SCU 34. This parameter consists of the port information for all the ports that are currently involved in this call and may be provided in one or more Port Information parameters;

Digits Collected: Indicates the digits collected information up to this point in the service call. Examples of this type of information include automatic number identification (ANI), authorization code (Authcode), personal identification number (PIN), account code, called number, international digits (i.e., the facility code, country code and the language digit), etc.;

Point In Call: Indicates the time when it was determined that the SCU 34 should be queried;

Access Type: Identifies the type of the port(s) and the trunk type;

Call Type: Indicates the type of call;

Bearer Capability: Indicating the bearer capability of the call;

Signaling Information: Includes information concerning either (i) an SS7 message if the port is a SS7/ISUP port, or (ii) a PRI message if the port is an ISDN/PRI port.

Flow Control Encountered: Indicates that flow control is active and identifies the source (PSN or SCU) that initiated the flow control.

Serving Translations Scheme: Identifies the serving translations scheme of the service call.

An Off-Hook event notification message is reported to the SCU 34 when a specified port goes off-hook or answers. Parameters for the Off-Hook event notification include:

Session ID: (described further below);

Instruction Tag: Contains a value indicating this is an asynchronous event notification (described further below);

Port Information: Identifies the port that went off-hook or answered; and

Signaling Information: Includes either (i) an answer message (ANM) if the port is a SS7/ISUP port, or (ii) a CONNECT message if the port is an ISDN/PRI port.

A On-Hook event notification message is reported to the SCU 34 when a specified port goes on-hook or is released. Parameters for the On-Hook event notification include:

Session ID: (described further below);

Instruction Tag: Contains a value indicating this is an asynchronous event notification (described further below);

Port Information: Identifies the port that went on-hook or released; and

Signaling Information: Includes either (i) release (REL) message if the port is a SS7/ISUP port, or (ii) disconnect (DISC) message if the port is an ISDN/PRI port.

A Route Not Available event notification message is reported to the SCU **34** when none of the trunk members (or ports) of a specified terminating trunk group are idle. This event notification is sent in response to the "Connect" primitive. Parameters for the Route Not Available event notification include:

Session ID: (described further below);

Instruction Tag: (described further below); and

Port Information: Identifies the port to which the terminator was supposed to be connected.

A Route Selected event notification message is reported to the SCU **34** when a valid terminating trunk group is identified and a member (port) of that terminating trunk group is found idle and that member is seized. This event notification is sent in response to the "Connect" primitive. Parameters for the Route Available event notification include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port to which the terminating trunk agency is connected; and

Port Information: Identifies the new terminating trunk agency/port.

A Signaling Event event notification message is reported to the SCU **34** when a port involved in a service call receives a signaling message on its peripheral. Parameters for the Signaling Event event notification may include:

Session ID: (described further below);

Instruction Tag: Contains a value indicating this is an asynchronous event notification (described further below);

Port Information: Identifies the port on which the signaling information was received; and

Signaling Information: Includes one of the following: (i) indicates the digits were outpulsed on the PTS trunk agent, (ii) any SS7 message (other than ANM and REL) if the port is a SS7/ISUP port, and (iii) any PRI message (other than CONNECT and DISC) if the port is an ISDN/PRI port.

A Tone Detected event notification message is reported to the SCU **34** when a specified tone/digit is detected on the given port. This event notification is sent in response to the "Monitor" primitive. Parameters for the Tone Detected event notification include:

Session ID: (described further below);

Instruction Tag: (described further below);

Port Information: Identifies the port on which the tone/digit was detected;

Tone Detected: Indicates the tone or digit that was detected.

3. Parameters

The following is a description of parameters which are generally applicable to primitive and/or event notification messages. Any parameter having application to specific message will be described below in relation to the respective primitive or event notification.

The Access Type parameter is utilized in the "New Call" event notification and contains the originating trunk information. It will be understood that the Access Type field for indicating the access type of the port may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0	Unused (Spare)
0 0 0 0 1	PTS FGD
0 0 0 1 0	SS7 FGD
0 0 0 1 1	DAL 4-wire
0 0 1 0 0	DAL 2-wire
0 0 1 0 1	PRI
0 0 1 1 0 to 1 1 1 1 1	Unused (Spare)

As will be appreciated, other Access types and bit patterns may be utilized to describe the access format of the call.

The Bearer Capability parameter contains the bearer capability information. This parameter includes a Bearer Capability field for indicating the speed and type of information that is to be carried by the call. It will be understood that the Bearer Capability parameter may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Not Used
0 0 0 1	Speech
0 0 1 0	64K data
0 0 1 1	64k X25
0 1 0 0	56k Data
0 1 0 1	Data Unit
0 1 1 0	64K Restricted
0 1 1 1	3.1 kHz
1 0 0 0	7 kHz
1 0 0 1	Voice Data
1 0 1 0	64K Rate Data
1 0 1 1 to 1 1 1 1	Spare values

As will be appreciated, other speed and information type descriptions may be included depending on the data communications utilized, and other bit patterns may also be used.

The Billing Information parameter contains the billing information for the port in the service call. This parameter may be used to update multiple billing record fields for a port. Parameter contents may include:

Port Information Field: Identifies the port whose billing records are to be updated. The external trunk group number and the trunk member number are specified in the port information.

Number of Billing Information field: Indicates the number of billing information sub-parameters to follow. Each billing information sub-parameter consists of the billing info type and the billing information contents.

Billing Information Type field: Indicates how to interpret the billing information contents. Each application may define its own set of types. The billing information contents are interpreted depending upon the type. The following are examples of billing information types:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Unknown
0 0 0 0 0 0 0 1	Call Reference ID

-continued

Bit Pattern	Indication
0 0 0 0 0 0 1 0 to 1 1 1 1 1 1 1 1	Spare Values

Billing Information Contents field: Includes the billing information to place in the billing record.

The Call Reference ID (CRID) parameter is used for the transport of a call reference identifier. The CRID is encoded in a variable length digits field with the digits encoded in telephony binary coded decimal (TBCD) format. The valid range of digits for the CRID parameter is 1 to 9 digits. Parameter contents may include:

Number of Digits field: Indicating the number of CRID digits. Valid values include 1 to 9.

1 to 9 Digit fields: Each containing one digit in the TBCD format (for dual tone multifrequency (DTMF), which may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Filler
0 0 0 1	Digit 1
0 0 1 0	Digit 2
0 0 1 1	Digit 3
0 1 0 0	Digit 4
0 1 0 1	Digit 5
0 1 1 0	Digit 6
0 1 1 1	Digit 7
1 0 0 0	Digit 8
1 0 0 1	Digit 9
1 0 1 0	Digit 0
1 0 1 1	*
1 1 0 0	#
1 1 0 1	D
1 1 1 0	E
1 1 1 1	F

The Call Type parameter contains the call type for the call when the PSN **28** determines that the call is to be controlled by the SCU **34**. The contents are definable by the applications. This parameter includes a Call Type field giving information on the type of call being made. Each application may define its own call type based on the application's needs. Values given below are examples of call types:

Bit Pattern	Indication
0 0 0 0 0 0	Undetermined
0 0 0 0 0 1	Onnet
0 0 0 0 1 0	Offnet
0 0 0 0 1 1	Public Speed
0 0 0 1 0 0	Private Speed
0 0 0 1 0 1	Hotline Speed
0 0 0 1 1 0	N00
0 0 0 1 1 1	Zero Plus - Onnet
0 0 1 0 0 0	Zero Plus - Offnet
0 0 1 0 0 1	INTOA
0 0 1 0 1 0 to 1 1 1 1 1 1	Spares

The Control Information parameter includes information relating to the action(s) to be taken if and when a primitive was successfully processed. Parameter contents include:

Trunk Group Number field: Identifies the trunk group number (plurality of bits);

Trunk Member Number field: Identifies the member of the trunk group number (plurality of bits); and

Send Instruction Completed Events field: A boolean for indicating whether the should or should not send any "Instruction Completed" event notification when an instruction is completed (1 bit, "1"—send all "Instruction Completed" events, "0" —do not send any "Instruction Completed" events).

The Destination Trunk Group parameter contains the external trunk group number of the trunk to which a call should be routed. The call terminates to an idle member of this trunk group. Parameter contents may include:

Trunk Group Number field: Identifies the external trunk group number (plurality of bits).

The Digit Collection parameter contains information required to collect digits on a specified port. Parameter contents may include:

Minimum digits field: Contains the minimum number of digits to collect.

Values may be between 0 to 45, for example:

Bit Pattern	Indication
0 0 0 0 0 0	Minimum number to collect is 0
0 0 0 0 0 1	Minimum number to collect is 1
. . .	
0 0 1 1 0 0	Minimum number to collect is 12
. . .	
1 0 1 1 1 0 to 1 1 1 1 1 1	Unused (Spare Values)

Maximum Digits field: Contains the maximum number of digits to collect. Values may be between 0 to 45, for example:

Bit Pattern	Indication
0 0 0 0 0 0	Maximum number to collect is 0
0 0 0 0 0 1	Maximum number to collect is 1
. . .	
0 1 0 1 0 0	Maximum number to collect is 20
. . .	
1 0 1 1 1 0 to 1 1 1 1 1 1	Unused (Spare Values)

End Delimiter field: Contains the delimiter and if the delimiter is dialed the digit collection will stop. If the delimiter is dialed as the very first digit then the digit collection will stop immediately. The delimiter digit values may include:

Bit Pattern	Indication
0 0 0	no delimiter specified
0 0 1	*
0 1 0	#
0 1 1	* and #
1 0 0 to 1 1 1	Spare

First Digit Timer field: Specifies the time to wait until the first digit is dialed. Valid values may be between 2 to 30 seconds, for example:

Bit Pattern	Indication
0 0 0 0 0 0	Unused (Spare)
0 0 0 0 0 1	Unused (Spare)
0 0 0 0 1 0	Timer Value = 2 second
. . .	
0 1 1 1 1 0	Timer Value = 30 seconds

-continued

Bit Pattern	Indication
0 1 1 1 1 1 to 1 1 1 1 1 1	Unused (Spare)

Inter Digit Timer field: Specifies the inter digit timer value, i.e., the time to wait when collecting the second and subsequent digits. Values may be between 1 to 20 seconds, for example:

Bit Pattern	Indication
0 0 0 0 0 0	Unused (Spare)
0 0 0 0 0 1	Timer Value = 1 second
. . .	
0 1 0 1 0 0	Timer Value = 20 seconds
0 1 0 1 0 1 to 1 1 1 1 1 1	Unused (Spare)

Discard Buffered Digits field: Is a boolean which if true indicates that if the PSN **28** had been buffering any digits prior to receiving this parameter, then the digits should be discarded and digit collection should start all over again.

The Digits Collected parameter contains the digits collected or received on the port/agent. The digits contained in this parameter are in the TBCD format. Also, included is a Count field which specifies the number of digits included in this parameter. Parameter contents may include:

Type of Digits field: Contains information about the digits collected, for example, encoded as given below. The type of digits is known when the collected digits are sent in the New Call event notification. When this parameter is sent in the Digits Collected event notification, a value of "Unknown" is used.

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Unknown
0 0 0 0 0 0 0 1	Called Party Address
0 0 0 0 0 0 1 0	Calling Patty Address (ANI)
0 0 0 0 0 0 1 1	Caller Interaction
0 0 0 0 0 1 0 0	Routing Number
0 0 0 0 0 1 0 1	Billing Number
0 0 0 0 0 1 1 0	Destination Number
0 0 0 0 0 1 1 1	Local Access and Transport Area (LATA)
0 0 0 0 1 0 0 0	Carrier Identification
0 0 0 0 1 0 0 1	Referral Number
0 0 0 0 1 0 1 0	True Billing Number
0 0 0 0 1 0 1 1	Alternate Preferred Carrier
0 0 0 0 1 1 0 0	Preferred INC
0 0 0 0 1 1 0 1	Primary Preferred Carrier
0 0 0 0 1 1 1 0	Personal ID Number (PIN)
0 0 0 0 1 1 1 1	Authorization Code
0 0 0 1 0 0 0 0	TCM
0 0 0 1 0 0 0 1	Second Alternate Preferred Carrier
0 0 0 1 0 0 1 0	Business Customer ID
0 0 0 1 0 0 1 1	Hop-off Office
0 0 0 1 0 1 0 0	Outpulse Number
0 0 0 1 0 1 0 1	Originating Station (DN)
0 0 0 1 0 1 1 0	MCCS Card Number
0 0 0 1 0 1 1 1	Account Code Number
0 0 0 1 1 0 0 0	COSOVE Number
0 0 0 1 1 0 0 1	Generic Digits Number
0 0 0 1 1 0 1 0	Dialed Digits Number
0 0 0 1 1 0 1 1	Facility Code
0 0 0 1 1 1 0 0	Country Code
0 0 0 1 1 1 0 1	Language Digit

-continued

Bit Pattern	Indication
0 0 0 1 1 1 1 0 to 1 1 1 1 1 1 1 1	Spare Values

Nature of Number field: Includes information about the nature of the number (digits) collected, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Not Applicable
0 0 0 0 0 0 0 1	International
0 0 0 0 0 0 1 0	National
0 0 0 0 0 0 1 1	Network Specific
0 0 0 0 0 1 0 0 to 1 1 1 1 1 1 1 1	Spare Values

Number of Digits field: Contains the number of digits that are sent in this parameter. This number may be as low as 1 and as high as 45.

Encoding Scheme field: Contains the scheme used to encode the digits that are sent in this parameter and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Unknown
0 0 0 1	Binary Coded Decimal (BCD)
0 0 1 0 to 1 1 0 1	Spare Values
1 1 1 0	Telephony Binary Coded Decimal (TBCD)
1 1 1 1	Spare

Numbering Plan field: Contains information about the numbering plan, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Unknown or Not Applicable
0 0 0 1	ISDN Numbering Plan (E.164)
0 0 1 0	Telephony Numbering Plan (E.163)
0 0 1 1	Data Numbering Plan (X.121)
0 1 0 0	Telex Numbering Plan (F.69)
0 1 0 1	Maritime Mobile Numbering Plan (E.120,211)
0 1 1 0	Land Mobile Numbering Plan (E.212,213)
0 1 1 1 to 1 1 0 1	Spare Values
1 1 1 0	Private
1 1 1 1	Reserved

1st to nth Digit field: Each containing one digit that may be encoded in the TBCD format (for dual tone multi-frequency (DTMF)).

The Digits To Outpulse parameter contains the digits to be outpulsed on a per-trunk signaling (PTS) trunk agent. Parameter contents may include: Number of Digits field: Contains the number of digits that are to be outpulsed on this port. Generally, the range for this field is 1 to 23.

1st to nth Digit field: Contains the n digits, and may be encoded in the TBCD format (for DTMF), or encoded as follows for multifrequency (MF):

Bit Pattern	Indication
0 0 0 0	Filler
0 0 0 1	Digit 1
0 0 1 0	Digit 2
0 0 1 1	Digit 3
0 1 0 0	Digit 4
0 1 0 1	Digit 5
0 1 1 0	Digit 6
0 1 1 1	Digit 7
1 0 0 0	Digit 8
1 0 0 1	Digit 9
1 0 1 0	Digit 0
1 0 1 1	Key pulse triple prime (KP3) and start translation triple prime (ST3P)
1 1 0 0	Key pulse prime (KPP) and start translation prime (STP)
1 1 0 1	Key pulse (KP) and STKP
1 1 1 0	Key pulse double prime (KP2) and start translation double prime (ST2P)
1 1 1 1	start translation (ST)

If multiple Digits To Outpulse parameters are contained in a message, then multiple streams of digits are outpulsed on the agent/port, each stream being contained in one parameter of type Digits to Outpulse.

The Digits Outpulsed parameter provides information about the digits that were outpulsed on a PTS trunk agent. This is especially useful in multi-stage outpulsing. This parameter is included in the "Signaling Event" event notification message to inform the SCU **34** that all the digits were outpulsed on the trunk agency. This parameter includes a Digits Outpulsed field which may be encoded as follows:

Bit Pattern	Indication
0 0 0	Unknown
0 0 1	all streams outpulsed
0 1 0 to 1 1 1	Spare

The Error Cause parameter contains the cause of an error that was detected in the PSN **28**. Parameter contents may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	NIL Error Cause
0 0 0 0 0 0 0 1	Header decode failure
0 0 0 0 0 0 1 0	Bad macro tag
0 0 0 0 0 0 1 1	Unrecognized primitive
0 0 0 0 0 1 0 0	Missing mandatory parameter
0 0 0 0 0 1 0 1	Mandatory parameter decode failure
0 0 0 0 0 1 1 0	Optional parameter decode failure
0 0 0 0 0 1 1 1	Parameter contents out of range
0 0 0 0 1 0 0 0	Primitive userclass mismatch
0 0 0 0 1 0 0 1	Maximum primitive exceeded
0 0 0 0 1 0 1 0	Missing mandatory Siginfo parameter
0 0 0 0 1 0 1 1	One or more agents in the primitive are not SN agents
0 0 0 0 1 1 0 0	Port not in table PSNROUTE
0 0 0 0 1 1 0 1	Agent not supported
0 0 0 0 1 1 1 0	Port down due to WARM restart
0 0 0 0 1 1 1 1	Primitive invalid for current port state
0 0 0 1 0 0 0 0	Unexpected message
0 0 0 1 0 0 0 1	STR not available (affects the Monitor primitive)
0 0 0 1 0 0 1 0	UTR not available
0 0 0 1 0 0 1 1	Conference circuit not available
0 0 0 1 0 1 0 0	No IDLE message
0 0 0 1 0 1 0 1	Primitive extension block not available
0 0 0 1 0 1 1 0	Scratchpad extension block not available
0 0 0 1 0 1 1 1	Software Resources unavailable
0 0 0 1 0 0 0 0	Message failure
0 0 0 1 0 0 0 1	Software error
0 0 0 1 0 0 1 0	Not minimum number ports to Bridge

-continued

Bit Pattern	Indication
0 0 0 1 0 0 1 1	Maximum ports to Bridge exceeded
0 0 0 1 1 1 0 0	Bearer capacity incompatible
0 0 0 1 1 1 0 1	Message index not in table PSNMSGIX
0 0 0 1 1 1 1 0	Unsupported signaling type
0 0 0 1 1 1 1 1	Duplicate message
0 0 1 0 0 0 0 0	Bad agent state
0 0 1 0 0 0 0 1	Termination failure
0 0 1 0 0 0 1 0	Abnormal exit
0 0 1 0 0 0 1 1	Message not playing
0 0 1 0 0 1 0 0	Tone duration unsupported
0 0 1 0 0 1 0 1	Prompt failure
0 0 1 0 0 1 1 0	Digit collection failure
0 0 1 0 0 1 1 1	Q764 protocol problem
0 0 1 0 1 0 0 0	Invalid duration gap
0 0 1 0 1 0 0 1	Unexpected FC message
0 0 1 0 1 0 1 0	Agent not in Table TRKGRP

The Error Type parameter contains information identifying whether a detected error is fatal or non-fatal, and may be encoded as follows:

Bit Pattern	Indication
0 0	Non Fatal Error
0 1	Fatal Error

Fatal errors are defined as errors that are so severe that they do not allow normal call processing to proceed on the port. If the error that is detected is non-fatal, then the PSN **28** does not take any action other than report this error to the SCU **34**. If the error that is detected is fatal, then the PSN **28** reports the error to the SCU **34** and takes down the associated port.

The Flow Control Information parameter contains programmable time periods for control of Duration and Gap. Duration refers to the maximum amount of time the flow controls is effective at the PSN **28** and Gap refers to the maximum rate at which "New Call" event notifications may be sent to the SCU **34** while the control is effective. This parameter includes:

Control Duration field: Contains the amount of time the flow control is effective at the PSN **28**, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Not Used
0 0 0 1	1 second
0 0 1 0	2 seconds
0 0 1 1	4 seconds
0 1 0 0	8 seconds
0 1 0 1	16 seconds
0 1 1 0	32 seconds
0 1 1 1	64 seconds
1 0 0 0	128 seconds
1 0 0 1	256 seconds
1 0 1 0	512 seconds
1 0 1 1	1024 seconds
1 1 0 0	2048 seconds
1 1 0 1 to 1 1 1 1	Spare

Control Gap field: Contains the maximum rate at which the New Call event notifications may be sent to the SCU **34**, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Remove Gap Control
0 0 0 1	0.1 second
0 0 1 0	0.3 second
0 0 1 1	0.5 second
0 1 0 0	1 second
0 1 0 1	2 seconds
0 1 1 0	5 seconds
0 1 1 1	10 seconds
1 0 0 0	15 seconds
1 0 0 1	30 seconds
1 0 1 0	50 seconds
1 0 1 1	80 seconds
1 1 0 0	120 seconds
1 1 0 1	300 seconds
1 1 1 0	600 seconds
1 1 1 1	Stop all calls

The Flow Control Encountered parameter contains information that informs the SCU **34** that the flow control is active and identifies the source that initiated the flow control, and may be encoded as follows:

Bit Pattern	Indication
0	Flow control initiated by SCU
1	Flow control initiated by PSN

The Instruction ID parameter contains the identifier of the primitive that was received from the SCU **34**. This parameter may be encoded, for example, as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Unknown
0 0 0 0 0 0 0 1	Bridge
0 0 0 0 0 0 1 0	Collect Digits & Report
0 0 0 0 0 0 1 1	Connect
0 0 0 0 0 1 0 0	Disconnect
0 0 0 0 0 1 0 1	Hold
0 0 0 0 0 1 1 0	Monitor
0 0 0 0 0 1 1 1	Mute
0 0 0 0 1 0 0 0	New Call Accepted
0 0 0 0 1 0 0 1	New Call Rejected
0 0 0 0 1 0 1 0	Play Message
0 0 0 0 1 0 1 1	Play Prompt, Collect Digits & Report
0 0 0 0 1 1 0 0	Query Port
0 0 0 0 1 1 0 1	Reconnect
0 0 0 0 1 1 1 0	Reset Switch
0 0 0 0 1 1 1 1	Set Billing Record
0 0 0 1 0 0 0 0	Set Port Service Information
0 0 0 1 0 0 0 1	Stop Message
0 0 0 1 0 0 1 0	Transmit SigInfo
0 0 0 1 0 0 1 1	Heartbeat
0 0 0 1 0 1 0 0	Query time of Day
0 0 0 1 0 1 0 1	Error Detected
0 0 0 1 0 1 1 0	Port Status
0 0 0 1 0 1 1 1	Flow Control
0 0 0 1 1 0 0 0 to 1 1 1 1 1 1 1 1	Spare Values

Instruction Tag parameter contains the tag associated with respective primitive instructions and event notifications. For primitives that are sent from the SCU **34**, the SCU **34** generates an Instruction Tag and sends it to the PSN **28** in this parameter. When a response for this primitive is generated by the PSN **28**, the PSN **28** includes this tag in the response so that the SCU **34** may correlate the response with the primitive request that it had sent earlier. For asynchronous event notifications (e.g., On-Hook, Off-Hook and Signalling Event) generated by the PSN **28**, the Instruction Tag is hard-coded to #01. For event notifications that are generated by the PSN **28** that require a reply from the SCU **34** (e.g., New Call, Query Port from the Audit application), the PSN **28** sends a nil Instruction Tag. This parameter may include, and may be encoded, as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	nil instruction tag
0 0 0 0 0 0 0 1	Asynchronous Event Notification from the PSN
0 0 0 0 0 0 1 0 to 1 1 1 1 1 1 1 1	Valid tags for primitives from the SCU

The Message Information parameter contains the Message ID and the Cycles/Tone Duration information. The Message ID may be a value that is used to index into a PSN message table which returns either an index into an Announcement table (if the message ID corresponds to an announcement) or a Tone table (if the message ID corresponds to a tone). This parameter may comprise two bytes to provide the following content:

Message ID field: Identifies the particular message.

Cycles field: Identifies the number of cycles the message should be played, if the Message ID corresponds to an announcement, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Play Announcement indefinitely
0 0 0 0 0 0 0 1	Play Announcement for 1 cycle
. . .	
0 0 0 1 1 1 1 0	Play Announcement for 30 cycles
0 0 0 1 1 1 1 1 to 1 1 1 1 1 1 1 1	Play Announcement indefinitely

Tone Duration field: Specifies the duration of the tone if the Message ID corresponds to a tone, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Play the Tone forever.
0 0 0 0 0 0 0 1	Invalid
0 0 0 0 0 0 1 0	Invalid
0 0 0 0 0 0 1 1	Play the Tone for 3 seconds
. . .	
1 1 1 1 1 1 1 1	Play the Tone for 255 seconds

The values for the cycles or tone duration may be set to any value when the Message Information parameter is sent in the Stop Message primitive. The

Stop Message primitive only cares about the message ID (i.e., to stop the appropriate message on the PSN) in the Message Information parameter.

The Monitor Mask parameter is a bitmap of monitor values. Each bit in this bitmap indicates the digit or tone to monitor or not monitor for a port. Each bit has two values—"0" indicating cancel the monitor or "1" indicating to start monitor. Parameter contents may include:

Tone Bitmap field: Containing three bits, each bit is a boolean for respective BBF, Octothorpe, and Asterisk monitoring. If the boolean is true then the port is monitored for the appropriate tone/digit. If the boolean is false then the appropriate tone/digit monitoring on that port is cancelled.

Octothorpe Duration field and Asterisk duration field: Indicates the duration in 100 milliseconds for which the octothorpe/asterisk has to be detected before it is reported as a valid tone/digit to the SCU **34**. For

example, a duration of 5 for an Asterisk implies that the user has to hold down the asterisk for 500 msecs before it is detected as a valid asterisk by the PSN **28** and reported to the SCU **34**. Note for BBF that the SF Tone Duration, the signalling type (MF/DTMF), the minimum digits to collect before a blue box fraud is declared and the partial dial timer values are all determined from a lookup table.

The Parameter ID parameter contains the identifier of the parameter (and the identifier of the sub-parameter if the parameter is composed of sub parameters). It is returned to the SCU **34** in case the decoding of the parameter resulted in an error. This parameter may include:

Parameter ID field: Used to identify the parameter in error, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Unknown
0 0 0 0 0 0 0 1	Bearer Capability
0 0 0 0 0 0 1 0	Billing Information
0 0 0 0 0 0 1 1	CallP Data Control
0 0 0 0 0 1 0 0	Call Reference Identifier
0 0 0 0 0 1 0 1	Control Information
0 0 0 0 0 1 1 0	Destination Trunk Group
0 0 0 0 0 1 1 1	Digit Collection
0 0 0 0 1 0 0 0	Digits Collected
0 0 0 0 1 0 0 1	Digits Outpulsed
0 0 0 0 1 0 1 0	Digits To Outpulse
0 0 0 0 1 0 1 1	Error Cause
0 0 0 0 1 1 0 0	Flow Control Information
0 0 0 0 1 1 0 1	Flow Control Encountered
0 0 0 0 1 1 1 0	Instruction ID
0 0 0 0 1 1 1 1	Instruction Tag
0 0 0 1 0 0 0 0	Message Information
0 0 0 1 0 0 0 1	Monitor Mask
0 0 0 1 0 0 1 0	Parameter ID
0 0 0 1 0 0 1 1	Port Count
0 0 0 1 0 1 0 0	Port Information
0 0 0 1 0 1 0 1	Port Service Information
0 0 0 1 0 1 1 0	Port Status
0 0 1 0 1 1 1 1	Reset Reason
0 0 0 1 1 0 0 0	Session ID
0 0 0 1 1 0 0 1	SigInfo Mask
0 0 0 1 1 0 1 0	Signaling Information
0 0 0 1 1 0 1 1	Switch ID
0 0 0 1 1 1 0 0	Time of Day
0 0 0 1 1 1 0 1	Tone Detected
0 0 0 1 1 1 1 0 to 1 0 0 0 0 0 0 1	Spare Values
1 0 0 0 0 0 1 0	UCS Points In Call
1 0 0 0 0 0 1 1	UCS STS
1 0 0 0 0 1 0 0 to 1 1 1 1 1 1 1	Spare Values

Sub Parameter ID field: Used to identify the parameter in error, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Unknown
0 0 0 0 0 0 0 1	Message Type
0 0 0 0 0 0 1 0	Digits Outpulsed
0 0 0 0 0 0 1 1	Digits To Outpulse
0 0 0 0 0 1 0 0	PTS Off-Hook
0 0 0 0 0 1 0 1	PTS On-Hook
0 0 0 0 0 1 1 0 to 1 1 1 1 1 1 1 1	Spare

The Point In Call parameter contains the point in the call when all criteria were met and the PSN **28** determined that the call is to be controlled by the SCU **34**. Parameter contents may include:

Point In Call field: Contains the point in the call, when the PSN determines that the call is a service call and needs to be controlled by the SCU. It is encoded as follows:

Bit Pattern	Indication
0 0 0 0 0	Not Used
0 0 0 0 1	Orig Null
0 0 0 1 0	Authorize Orig Attempt
0 0 0 1 1	Collect Information
0 0 1 0 0	Analyze Information
0 0 1 0 1	Select Route
0 0 1 1 0	Authorize Call Setup
0 0 1 1 1	Send Call
0 1 0 0 0	Orig Alerting
0 1 0 0 1	Orig Active
0 1 0 1 0	Orig Suspended
0 1 0 1 1	Term Null
0 1 1 0 0	Authorize Termination
0 1 1 0 1	Select Facility
0 1 1 1 0	Present Call
0 1 1 1 1	Term Alerting
1 0 0 0 0	Term Active
1 0 0 0 1	Term Suspended
1 0 0 1 0 to 1 1 1 1 1	Spare Values

Trigger Criteria field: It is encoded as follows:

0 0 0 0 0 0 0 0	Feature activator
0 0 0 0 0/ 0 0 1	Vertical Service Code
0 0 0 0 0 0 1 0	Customized Access
0 0 0 0 0 0 1 1	Customized Intercom*
0 0 0 0 0 1 0 0	NPA
0 0 0 0 0 1 0 1	NPA_NXX
0 0 0 0 0 1 1 0	NXX
0 0 0 0 0 1 1 1	NXX_XXXX
0 0 0 0 1 0 0 0	NPA_NXXXXXX*
0 0 0 0 1 0 0 1	Country_Code_NPA_NXXXXXX*
0 0 0 1 0 0 0 0	Offhook Immed
0 0 0 1 1 0 0 0	Net Busy
0 0 0 1 1 0 1 1	Orig Called Party Busy
0 0 0 1 1 1 0 1	Orig No Answer
0 0 1 0 0 0 0 0	Orig Feature Activator
0 1 1 0 0 0 0 0	Channel Setup PRI CLID
0 1 1 0 0 0 0 1	Channel Setup PRI Addr
0 1 1 0 0 0 1 0	Channel Setup PRI N00
0 1 1 0 0 0 1 1	Channel Setup PRI Intl
0 1 1 0 0 1 0 0	Specific Digit String Info*
0 1 1 0 0 1 0 1	Specific Digit String ANI*
0 1 1 0 0 1 1 0	Specific Digit String N00*
0 1 1 0 0 1 1 1	Specific Digit String CIC
0 1 1 0 1 0 0 0	Shared Interoffice CIC
0 1 1 0 1 0 0 1	Shared Interoffice Info.
0 1 1 0 1 0 1 0	Shared Interoffice ANI
0 1 1 0 1 0 1 1	Shared Interoffice Addr
0 1 1 0 1 1 0 0	Shared Interoffice N00
0 1 1 0 1 1 0 1	Shared Interoffice Intl
0 1 1 0 1 1 1 0 to 1 1 1 1 1 1 1 1	Unused (Spare)

The Port Information parameter contains the port information for an agent (or call). Parameter contents include:

Trunk Group Number field: Identifies the trunk group number (plurality of bits); and

Trunk Member Number field: Identifies the member of the trunk group number (plurality of bits).

The Port Service Information parameter contains port service information and Service Programming Interface (SPI) information. The SCU **34** provides this information to the PSN **28**. The port service information includes the return address used to send the event notification messages back to the SCU **34** (to the right address and port). The SPI number identifies the SPI version for the specified agent (or call). Parameter contents may include:

Port Information field: Contains the trunk group number (plurality of bits) and trunk member number (plurality of bits) of the port whose port service information is to be updated. The external trunk group number and the trunk member number are specified in the port information field.

Port Service Information field: Contains the port service information (e.g. return address) sent by the SCU **34** which is eventually used to return the event notification messages. This field may be a table of 4 bytes; for example, an return address of 47.122.64.153 is stored as 4 bytes—byte **1** is 47, byte **2** is 122, byte **3** is 64 and byte **4** is 153.

Port Number field: Contains the transport layer port number associated with the port service information.

SPI field: Contains the SPI version number for the specified agent.

Generally ranging from 1–15 (4 bits).

The Port Status parameter contains the status of the port/agent. It also contains information which indicates whether the port is currently being controlled by the SCU **34**. Parameter contents may include:

Agent Status field: Contains the agent status of the port, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0	Idle
0 0 0 1	Seized
0 0 1 0	Answered
0 0 1 1	ManBusy
0 1 0 0	Lockout
0 1 0 1	System Busy
0 1 1 0	PM Busy
0 1 1 1	Unknown
1 0 0 0 to 1 1 1 1	Spare Values

Port In Service Call field: Indicates if the port is not currently a part of a service call or the port that is currently being serviced by the SCU **34**. It may be encoded as:

Bit	Indication
0	Port not a part of a service call
1	Port is a part of a service call.

Connection Bitmap field: Is a bitmap where each bit indicates the connection status of the agent. The bits are used to represent the connection states as specified below, and may be encoded as follows:

Bit 0: Muted	
Values:	0 = port is un-muted, 1 = port is muted
Bit 1: Held	
Values:	0 = port is not held, 1 = port is held
Bit 2: Collecting Digits	
Values:	0 = port is not collecting digits 1 = port is in the process of collecting digits

-continued

Bit 3: Monitoring for Tones	
Values:	0 = port is not monitoring for tones 1 = port is monitoring for tones
Bit 4: Listening to Message	
Values:	0 = port is not listening to a message 1 = port is listening to message
Bit 5: Linked to an Agent	
Values:	0 = port is not linked to another agent 1 = port is linked to another agent
Bit 6: Bridged to multiple agents	
Values:	0 = port is not bridged to two or more agents 1 = port is bridged to two or more agents
Bit 7: Spare	

The Reset Reason parameter contains the reason why a restart was performed on the PSN **28**. It informs the SCU **34** of the type of reset that is required at the SCU **34**. This parameter may be encoded as follows and may include:

Bit Pattern	Indication
0 0 0 0	Unknown
0 0 0 1	Warm Restart performed on the PSN
0 0 1 0	Cold Restart performed on the PSN
0 0 1 1	Reload Restart performed on the PSN
0 1 0 0	PSN has just come into service
0 1 0 1	Arbitrator Heartbeat has failed
0 1 1 0 to 1 1 1 1	Spare Values

The Serving Translation Scheme parameter contains the serving translations scheme (STS) of the service call. This information is sent to the SCU **34** in the "New Call" event notification message. Parameter contents include:

Count field: Includes the number of digits in the STS, generally 1 to 3.

Digits 1, 2, 3 field: Depending on the number in the Count field, this field contains the digits (1 to 3), and may be encoded in the TBCD format.

The Session ID parameter contains a session identifier (ID). The session ID is generated by the SCU **34** to associate the different ports involved in a service call to a particular session. Upon receipt of the Session ID parameter, the PSN **28** does not perform any error checking. It merely re-transmits this value in the event notification message. Parameter contents include:

SCU ID field: Contains a unique one byte ID generated by the SCU **34**.

Session ID field: Contains a unique 3 byte ID generated by the SCU **34**.

The Signaling Information parameter contains the signalling information in the standard format that is applicable to the signaling type of the port. This parameter may be used to receive/send signalling information from/to the SCU **34**. Depending upon the primitive/event notification in which this parameter is sent/received and the signaling type of the associated port, its contents are different. In general, the contents contain parameters that are encoded in the standard format. For PTS agents, there are no standards used. For SS7 agents, the parameters are encoded based on the TR444 and GR394, and for PRI agents the TR1268 and ITU-T Q.931 is used to define the parameters. Signaling Information parameter contents may include:

Signaling Type field: Contains the type of signaling that the port is using, and may be encoded as follows:

Bit Pattern	Indication
0 0	PTS (4 Wire)
0 1	PTS (2 Wire)
1 0	SS7
1 1	PRI

Message Contents field: Contains the message contents in the standard format (SS7 or PRI). The appropriate content in respect of the SS7 and PRI types of signaling is identified in the primitives and event notifications, having the Signaling Information parameter.

PTS Message Type field: Contains the message type for a PTS message, and may be encoded as follows:

Bit Pattern	Indication
0 0 0 0 0 0 0 0	Unknown
0 0 0 0 0 0 0 1	Digits To Outpulse
0 0 0 0 0 0 1 0	Digits To Outpulse With Bearer Capability
0 0 0 0 0 0 1 1	PTS Off-Hook
0 0 0 0 0 1 0 0	PTS On-Hook
0 0 0 0 0 1 0 1	Digits Outpulse
0 0 0 0 0 1 1 0 to 1 1 1 1 1 1 1 1	Spare

PTS SigInformation Message field: Contains the PTS SigInformation message and depends on the PTS Message Type.

The SigInfo Mask parameter allows the SCU **34** to control which of the optional SS7/PRI messages are reported to the SCU **34** in the "Signaling Event" event notification message. This parameter consists of a bit map where each bit indicates the connection status of the agent, and may be encoded as follows:

Bit 0: ACM/Alert	
Values:	0 = Do not send ACM (for SS7 port) or Alert (for PRI port) message to the SCU. 1 = Send ACM (for SS7 port) or Alert (for PRI port) message to the SCU.
Bit 1: COT/Call Proc	
Values:	0 = Do not send COT (for SS7 port) or Call processing (for PRI port) message to the SCU. 1 = Send COT (for SS7 port) or Call processing (for PRI port) message to the SCU.
Bit 2: CPG/FAC	
Values:	0 = Do not send CPG (for SS7 port) or FAC (for PRI port) message to the SCU. 1 = Send CPS (for SS7 port) or FAC (for PRI port message to the SCU.
Bit 3: FAA	
Values:	0 = Do not send FAA (for SS7 port) message to the SCU. 1 = Send FAA (for SS7 port) message to the SCU.
Bit 4: FAR/Progress	
Values:	0 = Do not send FAR (for SS7 port) or Progress (for PRI port) message to the SCU. 1 = Send FAR (for SS7 port) or Progress (for PRI port message to the SCU.

-continued

Bit 5: FRJ/RLC	
Values:.	0 = Do not send FRJ (for SS7 port) or RLC (for PRI port) message to the SCU. 1 = Send FRJ (for SS7 port) or RLC (for PRI port message to the SCU.
Bit 6: PAM	
Values:	0 = Do not send PAM (for SS7 port) message to the SCU. 1 = Send PAM (for SS7 port) message to the SCU.
Bit 7: RES	
Values:	0 = Do not send RBS (for SS7 port) message to the SCU. 1 = Send RES (for SS7 port) message to the SCU.
Bit 8: SUS	
Values:	0 = Do not send SUS (for SS7 port) message to the SCU. 1 = Send SUS (for SS7 port) message to the SCU.
Bit 9: Spare	

The Switch ID parameter contains the switch identifier of the PSN **28**. A valid range may be 0 to 127 (1 byte (8 bits)).

The Time of Day parameter contains the time of the day and is returned in the "Current Time of Day" event notification to the SCU **34**. Parameter contents may include:

Year field: Contains the year.

Day In The Year field: Contains the day of the year.

Hour field: Contains the hour,

Minute field: Contains the minute.

Second field: Contains the second.

Ahead Of GMT field: Contains a boolean that if true, this field indicates that the switch time is ahead of the Greenwich Mean Time (GMT). The switch time is behind the GMT if this field is set to false.

Difference From GMT field: Contains the time difference between the time specified by the "hour, minute and second" fields above and the Greenwich Mean Time (GMT).

The Tone Detected parameter contains the tone or digit that was detected on a given port/agent by the PSN **28**. Its contents may be encoded as follows:

Bit Pattern	Indication
0 0 0	tone/digit monitoring aborted
0 0 1	asterisk detected
0 1 0	octothorpe detected
0 1 1	SF tone detected
1 0 0	BBF tone detected
1 0 1	BBF digits detected
1 1 0	BBF not possible on this port
1 1 1	Unused (Spare)

The above-described parameters describe one embodiment of the parameters utilized in the Service Programming Interface (SPI) of the present invention. As will be appreciated, other parameters may be used as desired, and other encoding schemes may be used by those of ordinary skill in the art.

4. Non-Call Related Primitives and Event Notifications

The following are non-call related primitives and event notifications which do not control a call topology or flow. They may be used for general non-call related activities, like administration and auditing purposes.

A Heartbeat primitive is sent to the PSN **28** from the SCU **34** to indicate the arbitrator is alive and well. If this message is not received within a predefined time, then the PSN **28** will start polling the arbitrator with an In Service message.

A Query Port primitive instructs the PSN **28** to query the status of a port on the PSN **28**. This instruction may also be sent from the PSN **28** to query the status of a port on the SCU **34**. This message is bidirectional. Parameters for the Query Port primitive include:

Session ID: (described above);

Instruction Tag: (described above); and

Port Information: Identifies the port whose status is requested.

A Query Time of the Day primitive instructs the PSN **28** to return the time of day. Parameters for the Query Port primitive include: Instruction Tag: (described above).

A Reset Switch primitive is sent to the PSN **28** by the SCU **34**, if the SCU **34** wants to reset certain ports that are being serviced. The ports to be reset are associated with a particular IP address. If the IP address information is missing, then all ports that are serviced by the PSN **28** are taken down. Parameters for the Reset Switch primitive include:

Instruction Tag: (described above);

IP Address Information: Contains the IP address information of any SCU **34** that is no longer in service and require that the ports serviced by the identified IP address be idled (e.g. torn down); and

Return IP Address Information: Contains the IP address information of the SCU to which an Instruction Completed response is sent.

A Set IP Address primitive instructs the PSN **28** to update either the arbitrator's IP address or a specified port's Port Service Information with the given data. The new Port Service Information is used to send event notifications and primitive responses for the port. This primitive may be received at any time during the call to update the appropriate Port Service Information for the port. Parameters include:

Instruction Tag: (described above); and

Port Service Information: Identifies the new return address used to report primitive responses and event notifications to the SCU **34** for the port or the arbitrator's IP address.

A Current Time of the Day event notification returns the current time-of-day tot he SCU **34**. This message is sent in response to the Query Time of Day primitive or when the time is changed on the PSN **28**. Parameters for the Current Time of the Day event notification include:

Current Time of Day: Contains the current time-of-day; and

Instruction Tag: (described above).

An In Service event notification is reported to the SCU **34** when the PSN **28** comes into service (e.g., after reset). Also, this message is sent to the SCU **34** when the arbitrator heartbeat fails. Parameters for the In Service event notification include:

Reset Reason: Identifies the reason the arbitrator is being requested (for example, because of a restart, heartbeat failure, or the system is coming into service).

A Port Status event notification is reported to the SCU **34** in response to the Query Port primitive from the SCU **34**. Parameters for the Port Status event notification include:

Session ID: (described above);

Instruction Tag: (described above);

Port Information: Identifies the port whose status is requested; and

Port Status: Identifies the status of the port.

E. Operation

In operation, an agent (i.e. a terminal or a trunk) associated with a call originating at a port in the programmable switch matrix **24** is made available to the SCU **34** by the PSN **28** entering a "server mode" of operation. In server mode, the agent participates in a client-server relationship with the SCU **34** whereby the call is under complete control of the SCU **34**. The SCU **34** provides all the instructions on how the agent is to proceed. The programmable switch matrix **24** (and PSN **28**) is not responsible for making deterministic decisions or taking actions upon a trunk which involves knowledge of services or predetermined reactions, as is the case with traditional call processing in typical switching systems. The service programming interface (SPI) (between the programmable switch matrix **24** and the SCU **34**) provides the SCU **34** with a set of call control primitives for the control and manipulation of calls within the programmable switch matrix **24**. Thereby, the SCU **34** has the ability to invoke within the PSN **28** such functions as digit collection, routing, playing announcements, call termination and bridging, specialized tone monitoring, out pulsing, and audio control over one or more parties when the call is in the server mode. From time to time, when external (i.e., peripheral) events occur at the programmable switch matrix **24** for any party that is involved in the SCU-controlled call, the SCU **34** is notified of these events by the PSN **28**. After server mode entry, the agent remains under the control of the SCU **34** until (a) the SCU **34** instructs the PSN **28** to disconnect the agent, (b) a peripheral event causes the agent to go on-hook, or (c) the agent is idled due to a fatal error at the PSN **28**. It is noted that the agent never goes back to in-switch traditional call processing once it enters the server mode.

At the programmable switch matrix **24**, an agent for a port (i.e., trunk) may be sent to the SCU **34** (thereby entering the server mode) upon encountering a trigger during processing of a call. Triggers may be employed through either a trigger database provisioned in the PSN **28** that contains SCU triggers and/or Carrier Advanced Intelligent Network (CAIN) triggers.

With respect to entering through the trigger database **102** (illustrated in FIGS. **4** and **5**), triggering is the process that the programmable switch matrix **24** uses to identify an agent requiring SCU **34** or SCP **26** handling. Triggering is based on points in call (PIC), trigger detection points (TDPs), call criteria and the trigger database datafill located in the programmable switch matrix **24**. A PIC is defined as a call processing event, examples of which are originate, and collect digits and analyze. PICs are subdivided or further qualified to provide more discrete definitions of particular points in the call. For example, the "originate" PIC is qualified to distinguish it as an off-hook origination. The "collect digits and analyze" PIC is qualified by the kind of digits that the programmable switch matrix **24** expects to collect at that particular point in the dialing plan, such as an authorization code, a personal identification number (PIN), or an account code. Each qualified PIC has associated with it a trigger detection point (TDP) which indicates the need to check a call's criteria for a trigger. Each trigger has its own set of criteria. When the criteria are met, the programmable switch matrix **24** (including the PSN **28**) performs the action defined by the trigger database, i.e., send the call to the SCU **34** or send the call to a treatment (e.g. SCP **26**).

During internal programmable switch matrix **24** switch call processing, TDPs are encountered. When a TDP is encountered, the PSN **28** accesses its trigger database tables with the call criteria available at that TDP. The trigger database **102** contains tables that contain the information required to assign, arm and deploy triggers. They define a set of criteria to be met and actions to be taken when those

criteria are met. Within the trigger database tables, call related information is checked against the trigger criteria. If the trigger criteria are met, call processing continues according to the action defined within the trigger database. If the trigger criteria are not met, the programmable switch matrix **24** continues controlling the call with traditional (conventional) switching system call processing.

In the present invention, the trigger database **102** tables contain two methods of triggering to the SCU **34**. One method includes Query immediately to the SCU (QuerySCU). QuerySCU suspends the normal switch call processing in the programmable switch matrix **24**, causes entry into the server mode, and sends a New Call event notification message to the SCU **34**. As will be appreciated, the trigger database **102** may contain numerous triggers to invoke the QuerySCU.

The second way a call agent can be sent to the SCU is via the CAIN SCP **26**. In a conventional fashion, the trigger database **102** determines that the call is a CAIN call and therefore, sends the call to the SCP **26**. The SCP **26** performs processing and determines that the call is an SCU call. The SCP **26** informs the PSN **28** through a Transaction Capabilities Application Protocol (TCAP) message containing predetermined encoding indicating the call should be connected to the SCU **34**. Upon receipt of the SCP "Connect to SCU" TCAP message, the programmable switch matrix **24** enters the server mode for that call and sends a New Call event notification message to the SCU **34**.

Server mode operation of the PSN **28** (and programmable switch matrix **24** for that call) is implemented by a finite state machine (FSM) around which the Service Programming Interface (SPI) protocol has been designed. The finite state machine comprises states, events, and allowable transitions between the states responsive to the events. For instance, the PSN **28** receives a primitive instruction (event) requiring some action be taken in respect of a certain port which is in a particular state. The results of performing this action typically results in transition of the port into another state. However, an event may or may not be valid depending upon the state of the port at that time. The finite state machine based implementation is well suited to define operation in such an event-response processing environment.

1. Finite State Machine

Now referring to FIG. **12***a*, **12***b*, and **12***c*, there is illustrated the finite state machine (FSM) **110** (illustrated in FIG. **5**) in accordance with the present invention. FIGS. **12***a* and **12***b* describe, the explicit and the implicit state transitions, respectively, within the PSN **28** in response to primitives received from the SCU **34**. FIG. **12***c* illustrates state transitions upon receipt of peripheral events. The combination of FIGS. **12***a–c* illustrate the complete port state transitions of the FSM **110**.

A port, in this particular embodiment of the FSM **110**, corresponds to a trunk member and generally to communication coupling with a party in a call. The port state may be defined by a combination of two states, namely an Agent state and a Connection state. Instructions (primitives) received from the SCU **34**, as well as the peripheral events on that port, are termed Events.

The Agent state describes the state of the port/agent at any given time. It does not describe the connection status of the agent. The port/agent may be in one of three Agent states: IDLE—the port is in an idle state; SEIZED—the port is in a seized and ringing state; and ANSWERED—the port is in a seized and answered state.

The Connection state is defined as the state of the connection that the agent/port is involved in at any given time. The port/agent may be in one of three Connection states: HELD—the port has no voice path between itself and the port to which it is connected (for example, in a port A to port B connection, A and B cannot hear each other); LINKED—the port has a two-way voice path between itself and the port to which it is connected (in a port A to port B connection, A and B can hear each other); and LISTENING—the port has one-way voice path between itself and the port to which it is connected (in a port A to port B connection, port A may be able to hear port B but port B cannot hear port A, etc).

As will be appreciated, the FSM **110** applies to any party in a service call (i.e., the originator, terminator, etc.). Hence, each port has an instance of the FSM **110** running on its behalf.

The state changes for a port depend upon the current state of that port. For example, when port A enters the server mode it is in an ANSWERED/HELD (agent/connection) state. When it is subsequently connected to port B, for instance, through an instruction sent by the SCU **34**, the state of port A changes to ANSWERED/LINKED; this is known as a state transition. The state of port B is SEIZED/LINKED and remains so until port B answers, and the state of port B then changes to ANSWERED/LINKED.

When an event is received for a port, depending upon the state of the port, a corresponding Action Processor is invoked. Action Processors are software procedures which when executed carry out the specific actions within the PSN **16** required by the instructions received from the SCU **34** or by peripheral events. Action processors control every aspect of the call processing at the programmable switch matrix **24** for calls in server mode.

Illustrated in FIG. **12***a* are explicit port transitions. An explicit state transition is a transition that occurs in direct response to the receipt of a primitive from the SCU **34**. The primitive may be received from the SCU **34** in connection with a port, when the port is in any port state (which is the combination of agent and connection states). Upon receipt of the primitive, one of the following actions occur. The primitive is valid for the port in its current state. Hence, the primitive is executed in this state. As a result of this execution, there may or may not be a state change for the port. Or, the primitive is invalid for the port in its current state. The primitive is not executed. The port remains in its current state. The SCU **34** is notified of the error that was detected in the Error Detected event notification message. It is noted that the Set Billing Record and Transmit SigInfo primitives do not cause any state changes and are valid in all but the IDLE/HELD state. The Query Port and Error Detected primitives are valid in all the states and also do not cause state changes. Hence, these four primitives have not been illustrated in FIG. **12***a*. Furthermore, the New Call is an event notification rather than an SCU primitive. This starts the first state transition out of the IDLE/HELD state for the port that enters the server mode.

Illustrated FIG. **12***b* are implicit state transitions. An implicit state transition is a transition that occurs as in indirect response to the receipt of a primitive from the SCU **34**. For example, in a two party call where parties A and B are linked (i.e., talking), a Disconnect primitive on port A causes an explicit state transition from the ANSWERED/LINKED state to the IDLE/HELD state for Party A, as shown in FIG. **12***a*. Moreover, the Disconnect primitive causes an implicit (or indirect) state transition for port B from ANSWERED/LINKED to ANSWERED/HELD state, as shown in FIG. **12***b*.

FIG. **12***c* illustrates state transitions upon receipt of peripheral events that are generated in the peripheral subsystem of the programmable switch matrix **24**, for example, the peripheral processors **88** of the programmable switch matrix **24** shown in FIG. **2**. These events are generally classified into the release, digits collected, outpulsing, tone detected and message played event type. A release event is any peripheral event that indicates the port has been released, disconnected or idled. An answer event indicates

that the port has answered or gone off-hook. A digits collected event reports the digits that were collected on the port. An outpulsing event indicates that digits and/or other signaling information is being transmitted/received on this port. A tone detected event reports the tone/digit that was detected on the port. A message played event reports that the announcement/tone connected to a port has finished playing.

After describing above the operation of the PSN **28** in terms of the FSM **110**, the following description together with reference to FIGS. **13**–**28** will illustrate the event/response/notification relationship between the PSN **28** and the SCU **34** with respect to the service programming interface message flow.

The SCU **34** transmits instructions (i.e., primitives/macros) to the PSN **38** which controls the flow of the call (as will be appreciated, a message may include a single primitive or a macro). In response to these instruction messages, the PSN **28** performs the requested action (through the programmable switch matrix **24**) and also transmits event notification messages to the SCU **34**. The PSN **28** may also transmit event notification messages in response to peripheral events that occur on a port in the programmable switch matrix **24** such that the SCU **34** may update its database. The message protocol is designed so that when an instruction message is transmitted from the SCU **34** to the PSN **28**, the PSN **28** responds with a reply (event notification message) for the instruction—immediately or after completing one or more actions on the PSN **28** (within the programmable switch matrix **24**). And when the PSN **28** transmits an event notification message to the SCU **34**, the SCU **34** may send a reply back in response to the event notification.

2. Message Flow

Now referring to FIG. **13**, there is illustrated a New Call event notification message flow. After the PSN **28** determines through the triggering process that the call requires (or desires) control by the SCU **34** (i.e., becomes a service call), a New Call event notification message is sent to the SCU **34**. In addition, an Event Timer is started. In response to the New Call event notification message, the SCU **34** sends the New Call Accepted primitive or the New Call Rejected primitive, and/or any other call control primitives. The New Call Rejected primitive is sent if the SCU **34** does not want to (or cannot) control the call, and wants the PSN **28** to route the call to the PSNF treatment. The New Call Accepted primitive is sent if the SCU **34** has accepted this call and will control the call thereafter. Any other call control primitives may be sent if the SCU **34** has accepted the call and will control the call thereafter. Upon receipt of any of these primitives, the PSN Event Timer is cancelled.

Now referring to FIG. **14**, there is illustrated a Connect primitive message flow. The SCU **34** sends a Connect primitive instruction to the PSN **28** when it wants to route a call to an idle trunk member. As a result of performing this instruction, the PSN **28** responds with a Route Selected or Route Not Available event notification message. The Route Selected event notification is sent if an idle member of the trunk group is located and the call is terminated onto this trunk member (i.e. port A is connected to port B, etc.). The Route Not Available event notification is sent if no member of the trunk group is found to be idle, i.e., a terminating trunk member is not found. As will be appreciated, the PSN **28** may also send an Error Detected event notification message if an error is detected (e.g., an error in the incoming instruction, an error due to unavailable resources, etc.). The Error Detected event notification may be sent in response to any primitive instruction for which an error may be detected.

Now referring to FIG. **15**, there is illustrated a Play Message primitive message flow. The Play Message primitive instruction is sent to the PSN **28** when the SCU **34** wants a message (i.e., an announcement or a tone) played to a specified port. The PSN **28** plays the message (using resources within the programmable switch matrix **24** or without the matrix, such as the MRU **36**, etc.) on the port and sends an Instruction Completed event notification message to the SCU **34**. When the message has finished playing, the PSN **28** also responds with a Message Played event notification message. The PSN **28** may also send an Error Detected event notification message if an error is detected.

Now referring to FIG. **16**, there is illustrated a message flow for the Collect Digits & Report primitive and the Play Prompt, Collect Digits & Report primitive. The SCU **34** sends the Collect Digits & Report or the Play Prompt, or a Collect Digits and Report primitive instruction to the PSN **34** for collection of a specific amount of digits on the given port. The latter is sent to play a message (i.e., a tone or an announcement) prior to the collection of the first digit on the port. The message played in this case is generally interruptible. Upon receipt of either of these primitives, the PSN **28** begins digit collection and sends back an Instruction Completed event notification message to the SCU **34**. After completing these instructions, the PSN responds with the Digits Collected event notification if the digit collection is successfully started and some/all the digits are collected or the digit collection has timed out. The PSN **28** may also send an Error Detected event notification message if an error is detected.

Now referring to FIG. **17**, there is illustrated a message flow for the Disconnect, Reconnect, Hold, Mute and Stop Message primitive instructions. The Disconnect primitive instruction is sent to the PSN **28** when the SCU **34** wants to disconnect (or remove) a given port from the service call. The Hold primitive instruction is sent to hold a given port and the Mute primitive instruction is sent to mute a given port. The Reconnect primitive instruction is sent when the SCU **34** wants to reestablish a connection between any two ports in the service call. The Stop Message primitive instruction is sent when the SCU **34** wants to stop playing the message that is currently being played on the specified port. After receiving any one of the above primitives, the PSN **28** has the option of responding with an Instruction Completed event notification message if the received instruction is successfully completed (e.g., the port is put on hold successfully). The PSN **28** may also send an Error Detected event notification message if an error is detected.

Now referring to FIG. **18**, there is illustrated a Bridge primitive message flow. The Bridge primitive instruction is sent to the PSN **28** when the SCU **34** wants to bridge (or conference in) several ports (and maybe even a message) in the service call. In response to the Bridge primitive, the PSN **28** has the option of sending to the SCU **34** the Instruction Completed event notification message upon successfully bridging all the ports (and optionally the message). If a message is bridged, then after the message has finished playing, the Message Played event notification message is sent to the SCU **34**. The PSN **28** may also send an Error Detected event notification message if an error is detected.

Now referring to FIG. **19**, there is illustrated a message flow for the Monitor primitive. The SCU **34** sends the Monitor primitive instruction to the PSN **34** when for monitoring a specified port for a given tone/digits. When the PSN **34** receives the Monitor primitive, the port is monitored (begins monitoring) for the specified tone/digits and returns an Instruction Completed event notification message to the SCU **34**. Upon completion of this instruction, the PSN **34** responds with the Tone Detected event notification message if the specified tone/digit is detected. The PSN **28** may also send an Error Detected event notification message if an error is detected (i.e., if there is a problem starting tone detection (e.g., STR is unavailable)).

Now referring to FIG. **20**, there is illustrated a message flow for the On-Hook, Off-Hook and Signaling Event event notification messages. When a port in a service call goes

on-hook, the PSN **28** sends the On-Hook event notification to the SCU **34**. When a port in a service call goes off-hook or answers, the PSN **28** sends an Off-Hook event notification to the SCU **34**. When a port in a service call receives signaling information on its peripheral from the network, it is reported to the SCU **34** in the Signaling Event event notification. The SCU **34** updates its internal information after receiving the On-Hook, Off-Hook, or Signaling Information event notifications. The SCU **34** generally does not send a reply in response to receiving these event notification messages.

Now referring to FIG. **21**, there is illustrated a Transmit SigInfo primitive message flow. The Transmit SigInfo primitive instruction is sent to the PSN **28** when the SCU **34** wants to transmit some signaling information on a specified port (to be sent over the network). The PSN **28** responds with the Instruction Completed event notification if the received instruction is successfully completed (i.e., the signalling info is successfully transmitted). The PSN **28** may also send an Error Detected event notification message if an error is detected.

Now referring to FIG. **22**, there is illustrated a Set Billing Record message flow. The Set Billing Record primitive instruction is sent to the PSN **28** when the SCU **34** wants to update the billing record of a given port in the service call. The PSN **28** updates the billing record associated with the given port and sends an Instruction Completed event notification message to the SCU **34**. However, if an error is detected, and the incoming request cannot be processed, then the Error Detected event notification message is returned to the SCU **34**.

Another way to update the billing record of a given port without sending the above primitive is to include the Billing Information parameter with other primitives that are sent for this port. The Billing Information parameter is optional in the primitives other than Set Billing Record primitive. For example the Billing Information parameter may be sent in the Hold primitive for the port. In that case, the port will not only be held, but its billing record will also be updated appropriately.

Now referring to FIG. **23**, there is illustrated a Query Port primitive message flow. The Query Port primitive instruction is sent to the PSN **28** when the SCU **34** wants to determine the status of a port in the PSN **28** (in the programmable switch matrix **24**). In response, the PSN **28** sends a Port Status event notification message containing the status of the port to the SCU **34**. The PSN **28** may also send an Error Detected event notification message if an error is detected. In addition, Query Port may also be sent by the PSN **28** to determine the status of the port on the SCU **34**. The SCU **34** returns the status of the port in the Port Status message.

Now referring to FIG. **24**, there is illustrated a message flow for the Query Time of the Day primitive. The Query Time of the Day primitive instruction is sent to the PSN **28** when the SCU **34** wants to know the time of the day from the PSN **28**. The PSN **28** responds with a Current Time of the Day event notification message containing information. The PSN **28** may also send an Error Detected event notification message if an error is detected. The PSN **28** also sends the Current Time of the Day asynchronously whenever the time is changed on the PSN **28**.

Now referring to FIG. **25**, there is illustrated a Set IP Address primitive message flow. The Set IP Address primitive instruction is sent to the PSN **28** when the SCU **34** wants to set either set the IP address (i.e. Port Service Information) of a given port in the service call or set the address of the arbitrator. The PSN **28** stores the IP address for the port to be used to send replies and responses to the SCU **34** in the future. The arbitrator address is used to send the New Call event notification for any port that is to be controlled by the SCU **34**. After updating the IP address for the port, the PSN **28** sends an Instruction Completed event notification message to the SCU **34**. The PSN **28** may also send an Error Detected event notification message if an error is detected.

Another way to update the IP address of a given port without sending the above primitive is to include the Port Service Information parameter with other primitives that are sent for this port. The Port Service Information parameter is optional in most primitives.

Now referring to FIG. **26**, there is illustrated a message flow for the In Service event notification. When the PSN **28** comes into service it sends the In Service event notification to the SCU **34**. This event notification message is also sent when the SCU **34** heartbeat fails or the PSN **28** is just back from a Cold/Reload restart. Upon receipt of this message, the SCU **34** returns the address of the SCU arbitrator using the Set IP Address primitive. The arbitrator is the initial point of contact for all service call ports. In other words, the New Call event notification message for a port is always sent to the arbitrator. Subsequent messages for this port may be routed to the arbitrator or to a new IP address. The new IP address is used if the SCU **34** had sent another Set IP Address primitive to update the IP address of this port.

Now referring to FIG. **27**, there is illustrated a message flow for the Heartbeat primitive. Periodically, the SCU **34** sends the Heartbeat primitive to the PSN **28** to indicate that all is well with the arbitrator. If this message is not received within a preset time within the PSN **28**, a heartbeat failure occurs and the In Service event notification is sent to the SCU **34** to begin polling for the arbitrator address.

Now referring to FIG. **28**, there is illustrated a Reset Switch primitive message flow. The Reset Switch primitive instruction is sent to the PSN **28** when the SCU **34** wants to reset all the ports that are serviced by certain "service providing units" on the SCU **34**. The PSN **28** may or may not idle all the appropriate ports and returns an Instruction Completed event notification message to the SCU **34**. The Error Detected event notification may be sent to the SCU **34** after a Warm restart on the PSN **34** when the PSN **34** wants to reset all the PSN ports that are in a non-talking state and controlled by the SCU **34**. The SCU **34** resets the status of all these ports to idle. For Cold/Reload restarts, the In Service event notification is sent to the SCU **34** (with the appropriate restart reason). The SCU **34** tags or marks all the ports that it is currently servicing on the PSN **28** as idle. Also, the PSN **28** takes down or idles all the ports that are involved in an SCU-controlled call.

Another aspect of the programmable service architecture is its IP addressing capability. The PSN **28** presents all New Call event notification messages to a single IP address. In order to facilitate the distribution of various services into various processors in the network, the Service Programming Interface protocol provides a field containing a return IP address. This address is used by the PSN **28** to route subsequent response messages (e.g. event notifications) from the PSN **28** to the SCU **34**. By employing this routing technique, a service may be executed over multiple SCUs without specific transfer logic in the PSN **28** to accommodate the multiple interfaces.

Now referring to FIGS. **29***a*, **29***b* and **29***c*, there is shown an example of a typical service call, including the steps involved (messages between PSN **28** and SCU **34**) and state transitions of the FSM **110**.

Although the service control platform **32** is shown in FIG. **1** as being external to the PSTN **20**, services may be created and deployed on a platform directly integrated with the various network elements that constitute the backbone PSTN **20**. A single programmable switch matrix **24** (and PSN **28**) may be controlled by one or more SCUs **34** allowing services to be identified by a first SCU (or other processing node) and then processed by a separate SCU.

Furthermore, the SCU **34** (or SCUs) may interface with multiple PSNs **28** (and multiple programmable switch matrices **24**).

Although the present invention and its advantages have been described in the foregoing detailed description and illustrated in the accompanying drawings, it will be understood by those skilled in the art that the invention is not limited to the embodiment(s) disclosed but is capable of numerous rearrangements, substitutions and modifications without departing from the spirit and scope of the invention as defined by the appended claims.

What is claimed is:

1. An apparatus for controlling and processing a service call in a telecommunications system comprising:

a programmable switch matrix, the programmable switch matrix comprising:

a plurality of ports, including a first port and a second port,

one or more predetermined triggers for detecting when processing of a service call associated with the first port is desired to be controlled externally,

a service node for outputting an event notification message to a service control unit in response to the detection of the service call and for receiving one or more primitives from the service control unit, and

call processing circuitry for connecting the first port to the second port in response to control commands of the one or more primitives received by the service node from the service control unit, the service control unit externally controlling the programmable switch matrix and processing of the service call; and

a first communications link between the programmable switch matrix and the service control unit.

2. An apparatus in accordance with claim **1** wherein the service control unit includes one or more service application software programs for execution to control processing of the service call.

3. An apparatus in accordance with claim **3** wherein the event notification message includes call information received from the service call and the service control unit uses the call information to identify and select an appropriate one of the one or more service application software programs to control processing of the service call.

4. An apparatus in accordance with claim **4** further comprising means for data interfacing between the programmable switch matrix and the service control unit.

5. An apparatus for controlling and processing a service call in a telecommunications system comprising:

a programmable switch matrix, the programmable switch matrix comprising:

a plurality of ports, including a first port and a second port,

one or more predetermined triggers for detecting when processing of a service call associated with the first port is desired to be controlled externally,

a service node for outputting an event notification message to a service control unit in response to the detection of the service call and for receiving one or more primitives from the service control unit, and

call processing circuitry for connecting the first port to the second port in response to control commands of the one or more primitives received by the service node from the service control unit, the service control unit externally controlling the programmable switch matrix and processing of the service call;

a first communications link between the programmable switch matrix and the service control unit;

a media resource unit for generating and outputting a message to at least one of the plurality of ports;

a second communications link between the media resource unit and the programmable switch matrix for carrying the message to the at least one of the plurality of ports; and

a third communications link between the service control unit and the media resource unit for controlling the output of the message from the media resource unit.

6. An apparatus for controlling and processing a service call in a telecommunications system comprising:

a service control unit having one or more application software programs for controlling call processing of the service call;

a programmable switch matrix, the programmable switch matrix comprising:

a plurality of ports, including a first port and a second port,

means for generating a service control unit service call request when call control processing of the service call received on the first port is desired to be controlled by the one or more application software programs within the service control unit, the service control unit located external to the programmable switch matrix,

means for generating and outputting an event notification message to the service control unit in response to the service call request, the event notification message including call information associated with the service call,

means for receiving one or more primitives from the service control unit, and

means for connecting the first port to the second port in response to control commands of the one or more received primitives, the service control unit externally controlling the programmable switch matrix and processing of the service call;

a first data link between the programmable switch matrix and the service control unit for carrying the one or more primitives from the service control unit to the programmable switch matrix and for carrying the event notification message from the service control unit to the programmable switch matrix;

a media resource unit for generating and outputting a message to at least one of the plurality of ports;

a second data link between the media resource unit and the programmable switch matrix for carrying the message to the at least one of the plurality of ports; and

a third data link between the service control unit and the media resource unit for controlling the output of the message from the media resource unit.

7. An apparatus in accordance with claim **6** wherein the call information of the service call comprises information for selecting a one of the application software programs to control processing of the service call.

8. An apparatus in accordance with claim **6** wherein the means for generating the service call request comprises a trigger database having one or more predetermined triggers.

9. An apparatus in accordance with claim **6** wherein the programmable switch matrix further comprises:

means for generating a service control point service call request when processing of the service call received on the first port is desired to be controlled by a service control point having a first database of information, the service control point located external to the programmable switch matrix and for controlling processing of the service call; and

means for generating and outputting a service control point message to the service control point in response

to the service call request, the service control point message including call information associated with the service call.

10. An apparatus in accordance with claim 9 wherein the service control point processes the call information and determines that the service call should be controlled by the service control unit, and thereafter sends a first message to the programmable switch matrix, via a fourth data link between the programmable switch matrix and the service control point, to generate the event notification message.

11. A method of controlling and processing a service call in a telecommunications system, comprising the steps of:

receiving a call on a first port of a switch matrix;

determining from call information associated with the received call that processing of the received call should be controlled by a service application program located externally to the switch matrix;

transmitting a new call event notification message to a service control unit notifying the service control unit that the received call should be controlled by the service application program located externally to the switch matrix, the service control unit located externally to the switch matrix and including the service application program;

executing the service application program to control the processing of the received call;

transmitting one or more primitives from the service control unit to the switch matrix; and

connecting the first port to a second port of the switch matrix in response to control commands of the one or more primitives.

12. A method in accordance with claim 11 further comprising the steps of:

generating a message and outputting the message to a first output; and

connecting the first output to the second port.

13. A method in accordance with claim 11 wherein the step of determining comprises the steps of:

comparing the call information of the received call with call criteria within a trigger database;

suspending in-switch conventional call processing of the received call within the switch matrix; and

entering a server mode wherein the switch matrix performs one or more actions for controlling call processing of the received call in response to one or more primitives received from the service control unit.

14. A method of controlling and processing a service call in a telecommunications system, comprising the steps of:

receiving a call on a first port of a switch matrix;

determining from call information associated with the received call that processing of the received call should be controlled by a service application program located external the switch matrix;

transmitting a new call event notification message to a service control unit notifying the service control unit that the received call should be controlled by the service application program located external the switch matrix, the service control unit located externally to the switch matrix and including the service application program;

executing the service application program to control the processing of the received call;

transmitting one or more primitives from the service control unit to the switch matrix;

connecting the first port to a second port of the switch matrix in response to control commands of the one or more primitives; and

transmitting, in response to the new event notification message, a new call accepted primitive to the switch matrix, the new call accepted primitive comprising a first set of data defining the first port and a second set of data defining a return address within the service control unit to which one or more subsequent event notification messages will be sent from the switch matrix.

15. A method in accordance with claim 11 wherein the step of connecting the first port to the second port further comprises the steps of:

receiving a connect primitive from the service control unit, the connect primitive comprising,

a first set of data defining the first port, and

a second set of data defining a destination trunk; and

connecting the first port to an idle port of the destination trunk, the idle port defining the second port.

16. A method of controlling and processing a service call in a telecommunications system, comprising the steps of:

receiving a call on a first port of a switch matrix, the received call including call information;

performing in-switch call processing of the received call until a trigger is detected;

triggering when the call information of the received call meets one or more predetermined trigger criteria within a trigger database;

sending a first data message to a service control unit in response to the triggering, the service control unit located external to the switch matrix, the first data message comprising,

a first set of data defining a type of the received call, and

a second set of data defining an address of the first port;

sending a second data message from the service control unit to the switch matrix notifying the switch matrix that the first data message was received and that processing of the received call will be controlled by the service control unit;

selecting a predetermined service application program to handle control processing of the received call in response to the call type data of the first message;

executing the selected service application program, the selected service application program generating one or more instructions for performing one or more actions within the switch matrix;

sending the one or more instructions to the switch matrix to control processing of the received call; and

connecting the first port to a second port of the switch matrix in response to the one or more instructions.

17. A method in accordance with claim 16 wherein the step of triggering comprises the steps of:

comparing the call information of the received call with call criteria within the trigger database;

suspending in-switch processing of the received call within the switch matrix; and

entering a server mode wherein the switch matrix performs one or more actions for controlling call processing of the received call in response to one or more instructions received from the service control unit.

18. A method in accordance with claim 17 further comprising the steps of:

generating a message and outputting the message to a first output; and

connecting the first output to the second port.

19. A method in accordance with claim 16 further comprising the step of:

sending a second data message from the service control unit to the switch matrix defining a return address within the service control unit to which one or more subsequent data messages will be sent from the switch matrix.

20. A service programming interface protocol for use between a switch matrix and an external service control unit having one or more service application programs for execution for controlling a service call received on one of a plurality of ports of the switch matrix, the interface protocol comprising:

a plurality of primitives for transmission from the service control unit to the switch matrix, the plurality of primitives comprising:

a bridge primitive for instructing the switch matrix to connect a plurality of predetermined ports to each other within the switch matrix,

a collect digits and report primitive for instructing the switch matrix to receive a specified number of multifrequency digits at a predetermined port and transmit the received digits to the service control unit,

a connect primitive for instructing the switch matrix to connect a predetermined first port of the switch matrix to another port of the switch matrix,

a disconnect primitive for instructing the switch matrix to disconnect a predetermined first port of the switch matrix from another predetermined port of the switch matrix, and

a new call accepted primitive for instructing the switch matrix that the processing of the service call will be controlled by the service control unit; and

a plurality of event notification messages for transmission from the switch matrix to the service control unit, the plurality of event notification messages comprising:

a digits collected event notification message for informing the service control unit that a specified number of digits have been collected from a predetermined port and the identity of the collected digits,

an instruction completed event notification message for informing the service control unit that the switch matrix has completed one or more actions in response to one or more primitives previously received from service control unit,

a new call event notification message for informing the service control unit that a triggering event has occurred for the received call and that the service call is to be controlled by the service control unit,

an off-hook event notification message for informing the service control unit that a predetermined port is off-hook, and

an on-hook event notification message for informing the service control unit that a predetermined port is on-hook.

21. The service programming interface protocol in accordance with claim **20** wherein the plurality of primitives further comprises:

a hold primitive for instructing the switch matrix to put a predetermined port on hold;

a monitor primitive for instructing the switch matrix to monitor a predetermined port for a predetermined number of digits or tones;

a mute primitive for instructing the switch matrix to mute or non-mute a voice path between a predetermined port and another predetermined port;

a play message primitive for instructing the switch matrix to connect a message to a predetermined port;

a reconnect primitive for instructing the switch matrix to re-establish a connection between a predetermined port and another predetermined port; and

a stop message primitive for instructing the switch matrix to stop playing a message to a predetermined port.

22. The service programming interface protocol in accordance with claim **21** wherein the plurality of primitives further comprises:

an error detected primitive for instructing the switch matrix that the service control unit has detected an error with a predetermined port;

a new call rejected primitive for instructing the switch matrix that the service call identified in the new call event notification message will not be controlled by the service control unit;

a play prompt, collect digits and report primitive for instructing the switch matrix to play a message to a predetermined port and collect digits on the predetermined port;

a set billing record primitive for instructing the switch matrix to update billing information for a predetermined port; and

a transmit siginfo primitive for instructing the switch matrix to transmit signaling information on a predetermined port.

23. The service programming interface protocol in accordance with claim **20** wherein the plurality of event notification messages further comprises:

an error detected event notification message for informing the service control unit that the switch matrix has detected an error on a predetermined port;

a message played event notification message for informing the service control unit that a message played to a predetermined port has ended;

a route not available event notification message for informing the service control unit that no ports of a predetermined trunk are idle;

a route selected event notification message for informing the service control unit that a port of a predetermined trunk has been found idle and seized;

a tone detected event notification message for informing that service control unit when a predetermined digit or tone is detected on a predetermined port; and

a signaling event event notification message for informing the service control unit that a predetermined port has received a signaling message and further identifying the signaling message.

24. A method of communicating, via a communications link, between a switch matrix and an external service control unit having one or more service application programs for controlling a service call received on one of a plurality of ports of the switch matrix, comprising the steps of:

instructing, by transmitting a bridge primitive from the service control unit to the switch matrix, the switch matrix to connect a plurality of predetermined ports to each other within the switch matrix;

instructing, by transmitting a collect digits and report primitive from the service control unit to the switch matrix, the switch matrix to receive a specified number of multifrequency digits at a predetermined port and transmit the received digits to the service control unit;

instructing, by transmitting a connect primitive from the service control unit to the switch matrix, the switch matrix to connect a predetermined first port of the switch matrix to another port of the switch matrix;

instructing, by transmitting a disconnect primitive from the service control unit to the switch matrix, the switch matrix to disconnect a predetermined first port of the

switch matrix from another predetermined port of the switch matrix;

instructing, by transmitting a new call accepted primitive from the service control unit to the switch matrix, the switch matrix that the processing of the service call will be controlled by the service control unit;

informing, by transmitting a digits collected event notification message from the switch matrix to the service control unit, the service control unit that a specified number of digits have been collected from a predetermined port and the identity of the collected digits;

informing, by transmitting an instruction completed event notification message from the switch matrix to the service control unit, the service control unit that the switch matrix has completed one or more actions in response to one or more primitives previously received from service control unit;

informing, by transmitting a new call event notification message from the switch matrix to the service control unit, the service control unit that a triggering event has occurred for the received call and that the service call is to be controlled by the service control unit;

informing, by transmitting an off-hook event notification message from the switch matrix to the service control unit, the service control unit that a predetermined port is off-hook; and

informing, by transmitting an on-hook event notification message from the switch matrix to the service control unit, the service control unit that a predetermined port is on-hook.

25. The method in accordance with claim **24** further comprising the steps of:

instructing, by transmitting a hold primitive from the service control unit to the switch matrix, the switch matrix to put a predetermined port on hold;

instructing, by transmitting a monitor primitive from the service control unit to the switch matrix, the switch matrix to monitor a predetermined port for a predetermined number of digits or tones;

instructing, by transmitting a mute primitive from the service control unit to the switch matrix, the switch matrix to mute or non-mute a voice path between a predetermined port and another predetermined port;

instructing, by transmitting a play message primitive from the service control unit to the switch matrix, the switch matrix to connect a message to a predetermined port;

instructing, by transmitting a reconnect primitive from the service control unit to the switch matrix, the switch matrix to re-establish a connection between a predetermined port and another predetermined port; and

instructing, by transmitting a stop message primitive from the service control unit to the switch matrix, the switch matrix to stop playing a message to a predetermined port.

26. The method in accordance with claim **24** further comprising the steps of:

instructing, by transmitting an error detected primitive from the service control unit to the switch matrix, the switch matrix that the service control unit has detected an error with a predetermined port;

instructing, by transmitting a new call rejected primitive from the service control unit to the switch matrix, the switch matrix that the service call identified in the new call event notification message will not be controlled by the service control unit;

instructing, by transmitting a play prompt, collect digits and report primitive from the service control unit to the switch matrix, the switch matrix to play a message to a predetermined port and collect digits on the predetermined port;

instructing, by transmitting a set billing record primitive from the service control unit to the switch matrix, the switch matrix to update billing information for a predetermined port; and

instructing, by transmitting a transmit siginfo primitive from the service control unit to the switch matrix, the switch matrix to transmit signaling information on a predetermined port.

27. The method in accordance with claim **24** further comprising the steps of:

informing, by transmitting an error detected event notification message from the switch matrix to the service control unit, the service control unit that the switch matrix has detected an error on a predetermined port;

informing, by transmitting a message played event notification message from the switch matrix to the service control unit, the service control unit that a message played to a predetermined port has ended;

informing, by transmitting a route not available event notification message from the switch matrix to the service control unit, the service control unit that no ports of a predetermined trunk are idle;

informing, by transmitting a route selected event notification message from the switch matrix to the service control unit, the service control unit that a port of a predetermined trunk has ben found idle and seized;

informing, by transmitting a tone detected event notification message from the switch matrix to the service control unit, the service control unit when a predetermined digit or tone is detected on a predetermined port; and

informing, by transmitting a signaling event event notification message from the switch matrix to the service control unit, the service control unit that a predetermined port has received a signaling message and further identifying the signaling message.

* * * * *

EXHIBIT L

US006799210B1

(12) **United States Patent**
Gentry et al.

(10) **Patent No.: US 6,799,210 B1**
(45) **Date of Patent: Sep. 28, 2004**

(54) **DYNAMIC ASSOCIATION OF ENDPOINTS TO MEDIA GATEWAY CONTROLLERS**

(75) Inventors: **William D. Gentry**, Raleigh, NC (US); **Jason G. Emery**, Apex, NC (US); **Samuel H. Christie, IV**, Cary, NC (US)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 765 days.

(21) Appl. No.: **09/598,867**

(22) Filed: **Jun. 21, 2000**

(51) **Int. Cl.**[7] **G06F 15/173**

(52) **U.S. Cl.** **709/223**; 709/226; 709/227; 709/229; 709/238; 370/352; 370/356; 370/401

(58) **Field of Search** 709/223, 226–230, 709/235–238, 244, 249–250; 370/229, 352–356, 400–401

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,449,251 B1 * 9/2002 Awadallah et al. 370/229
6,515,985 B2 * 2/2003 Shmulevich et al. 370/356
6,603,760 B1 * 8/2003 Smyk 370/352
6,611,516 B1 * 8/2003 Pirkola et al. 370/352
6,631,416 B2 * 10/2003 Bendinelli et al. 709/227
6,671,263 B1 * 12/2003 Potter et al. 370/401
6,674,713 B1 * 1/2004 Berg et al. 709/227
6,680,952 B1 * 1/2004 Berg et al. 370/401

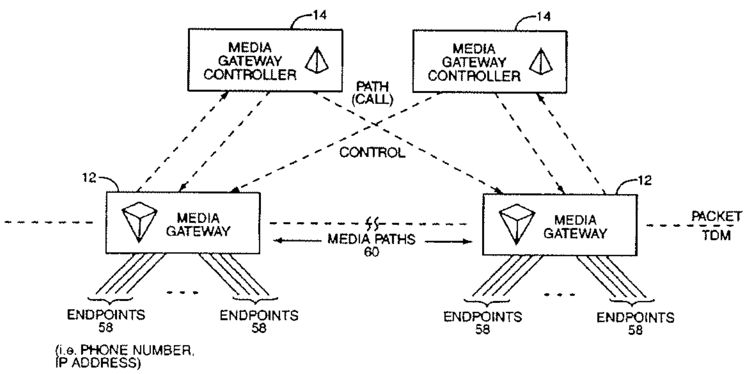
* cited by examiner

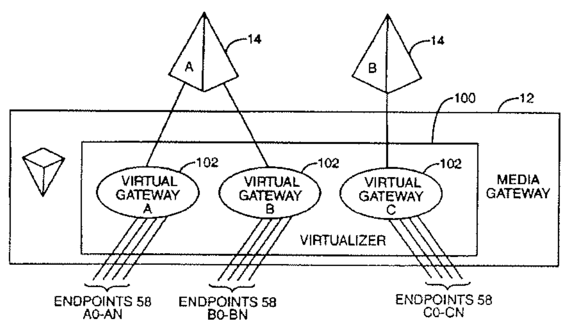
Primary Examiner—Bharat Barot
(74) *Attorney, Agent, or Firm*—Withrow & Terranova, PLLC

(57) **ABSTRACT**

The present invention provides an architecture for a media gateway to identify and register with multiple media gateway controllers for various types of voice and data services, along with having the media gateway appear to each of these media gateway controllers as a single, dedicated control entity. A logical layer, referred to as a virtualizer, is associated with each media gateway and appears to the media gateway as a single media gateway controller. To each media gateway controller the virtualizer supports, the virtualizer appears as a single media gateway. In essence, the virtualizer is a protocol manager and message router. The virtualizer supports the registration of multiple media gateways and then creates virtual gateways based on requirements of the media gateway controllers serving these virtual gateways. Preferably, subscribers associated with endpoints handled by the media gateway are grouped into a virtual gateway being served by a select media gateway controller or group thereof.

28 Claims, 9 Drawing Sheets





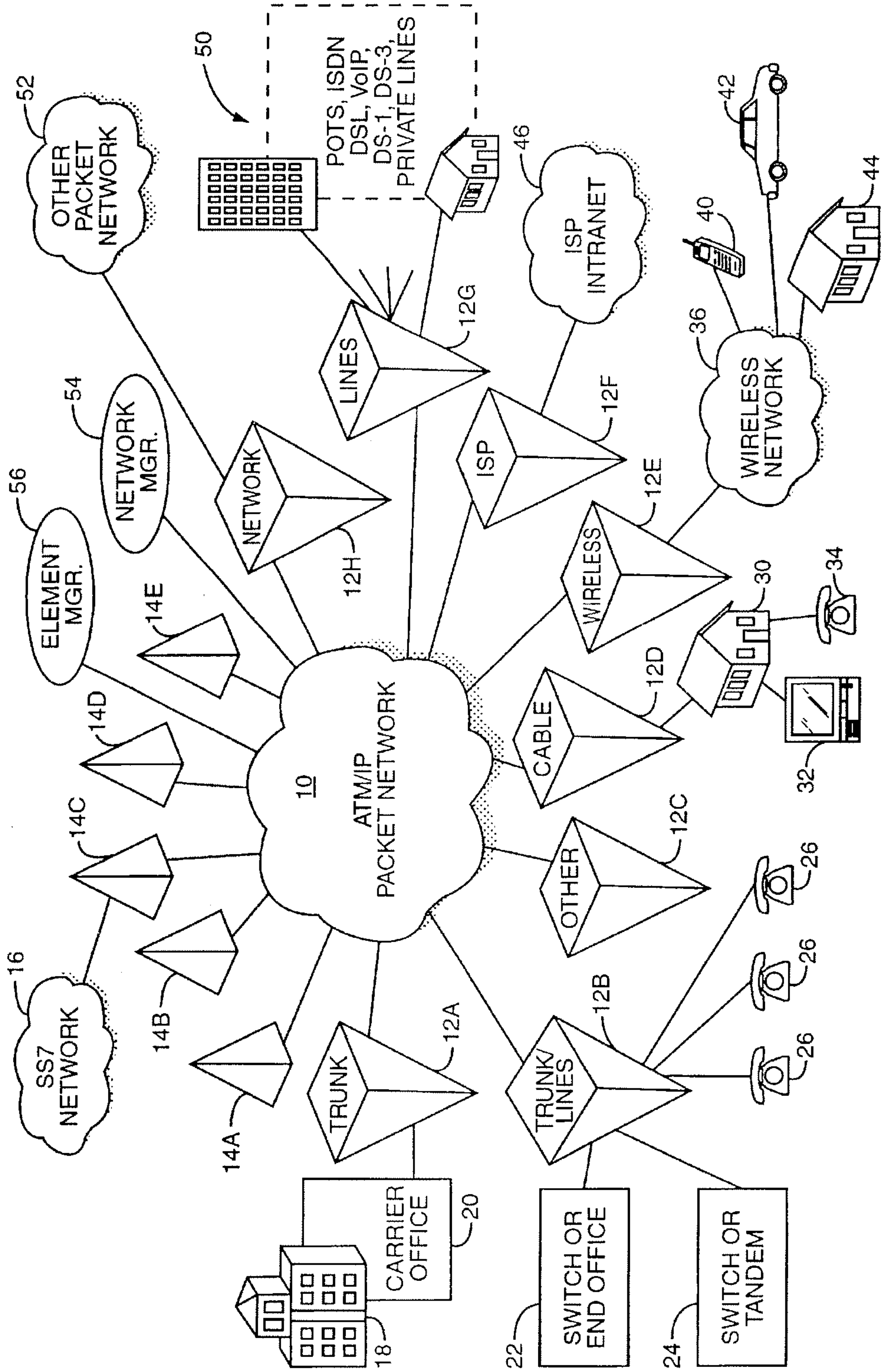

ATM/IP
PACKET NETWORK
10
TRUNK
12A
TRUNK/
LINES
12B
OTHER
12C
CABLE
12D
WIRELESS
12E
ISP
12F
LINES
12G
NETWORK
12H
14A
14B
14C
14D
14E
SS7
NETWORK
16
18
CARRIER
OFFICE
20
SWITCH OR
END OFFICE
22
SWITCH OR
TANDEM
24
26
26
26
30
32
34
WIRELESS
NETWORK
36
40
42
44
ISP
INTRANET
46
50
POTS, ISDN
DSL, VoIP,
DS-1, DS-3,
PRIVATE LINES
OTHER
PACKET
NETWORK
52
NETWORK
MGR.
54
ELEMENT
MGR.
56

FIG. 1

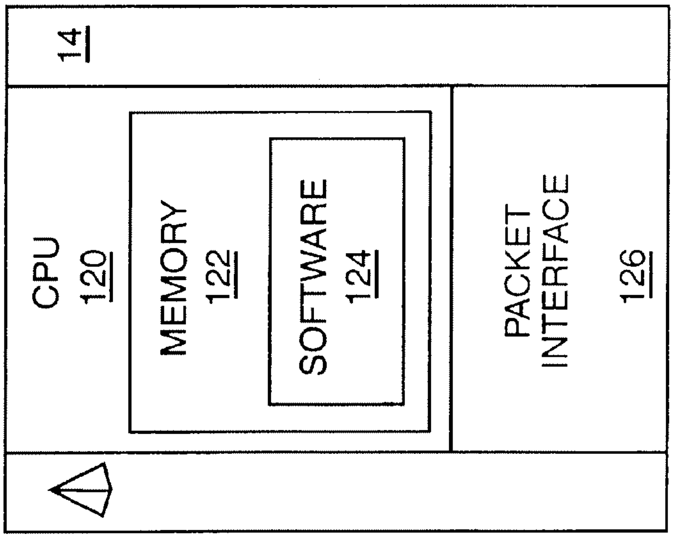

14
CPU 120
MEMORY 122
SOFTWARE 124
PACKET INTERFACE 126

FIG. 3

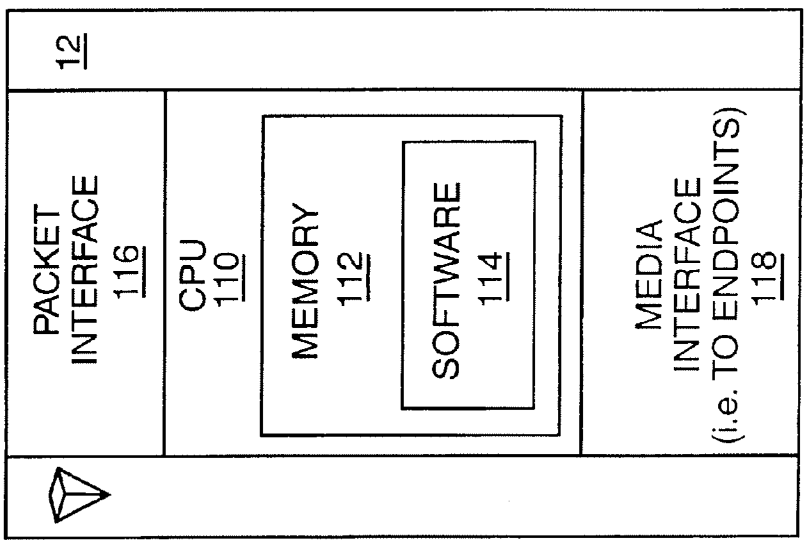
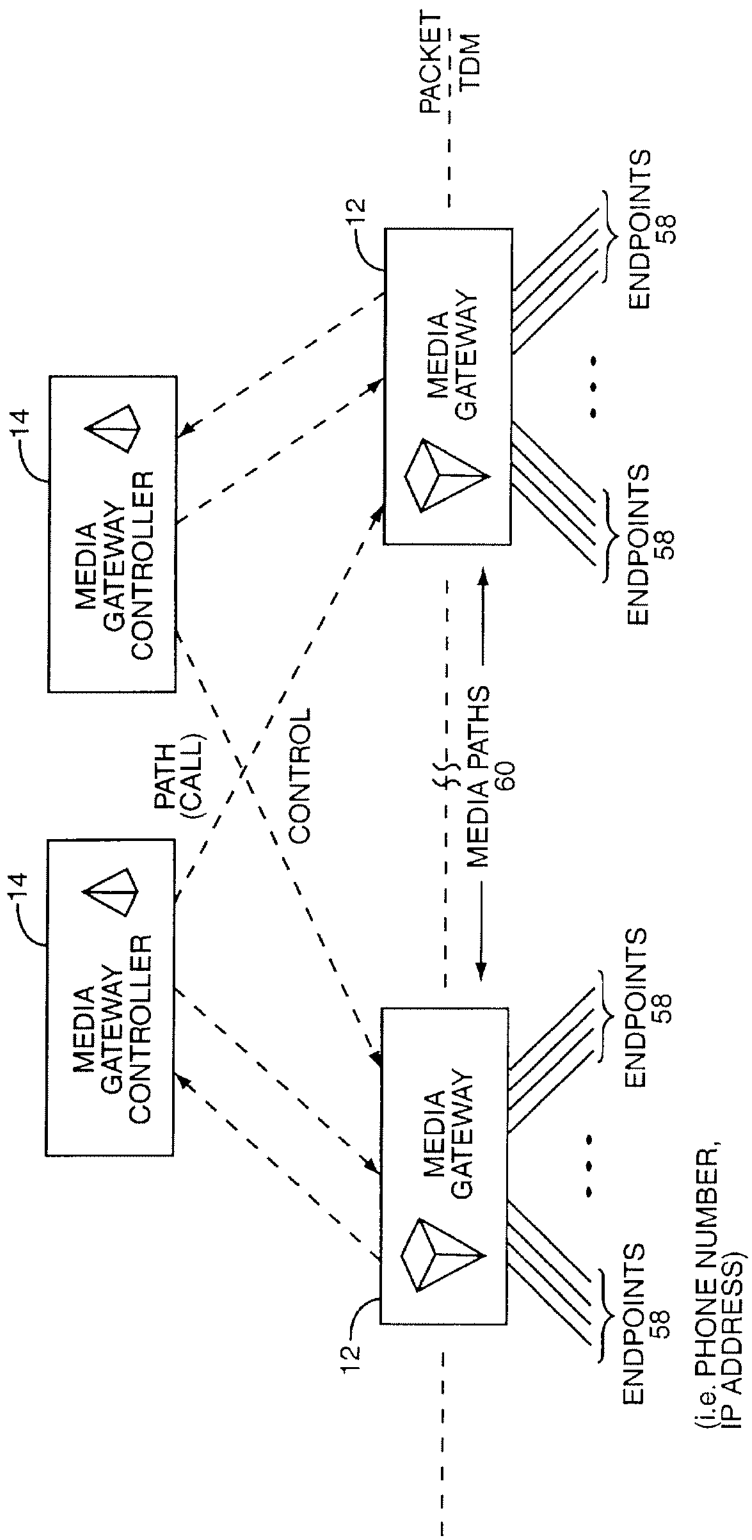

12
PACKET INTERFACE 116
CPU 110
MEMORY 112
SOFTWARE 114
MEDIA INTERFACE (i.e. TO ENDPOINTS) 118

FIG. 2



PACKET
TDM
14
MEDIA GATEWAY CONTROLLER
14
MEDIA GATEWAY CONTROLLER
12
MEDIA GATEWAY
12
MEDIA GATEWAY
ENDPOINTS 58
ENDPOINTS 58
ENDPOINTS 58
ENDPOINTS 58
PATH (CALL)
CONTROL
MEDIA PATHS 60
(i.e. PHONE NUMBER, IP ADDRESS)

FIG. 4

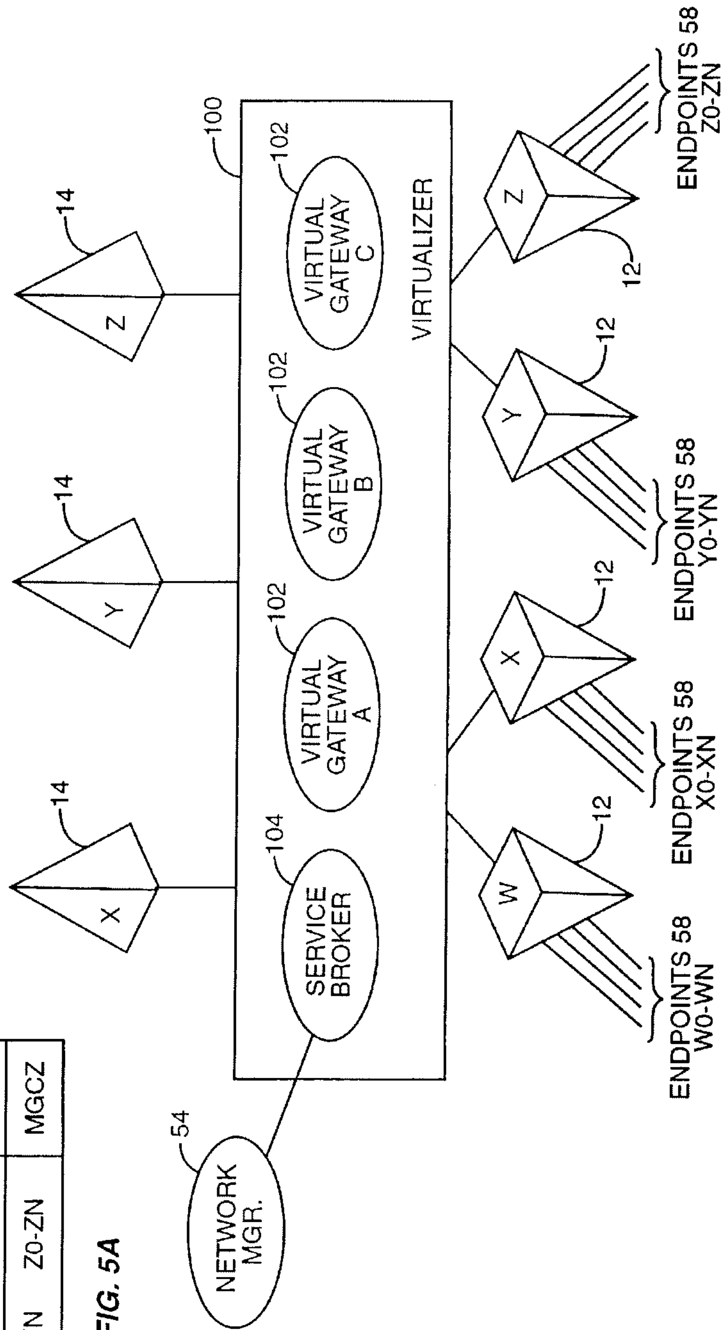


FIG. 5

MAPPING

VG	ENDPOINTS	MGC
VGA	W0-WN X1-X3	MGCX
VGB	X4-XN	MGCY
VGC	Y0-YN Z0-ZN	MGCZ

FIG. 5A

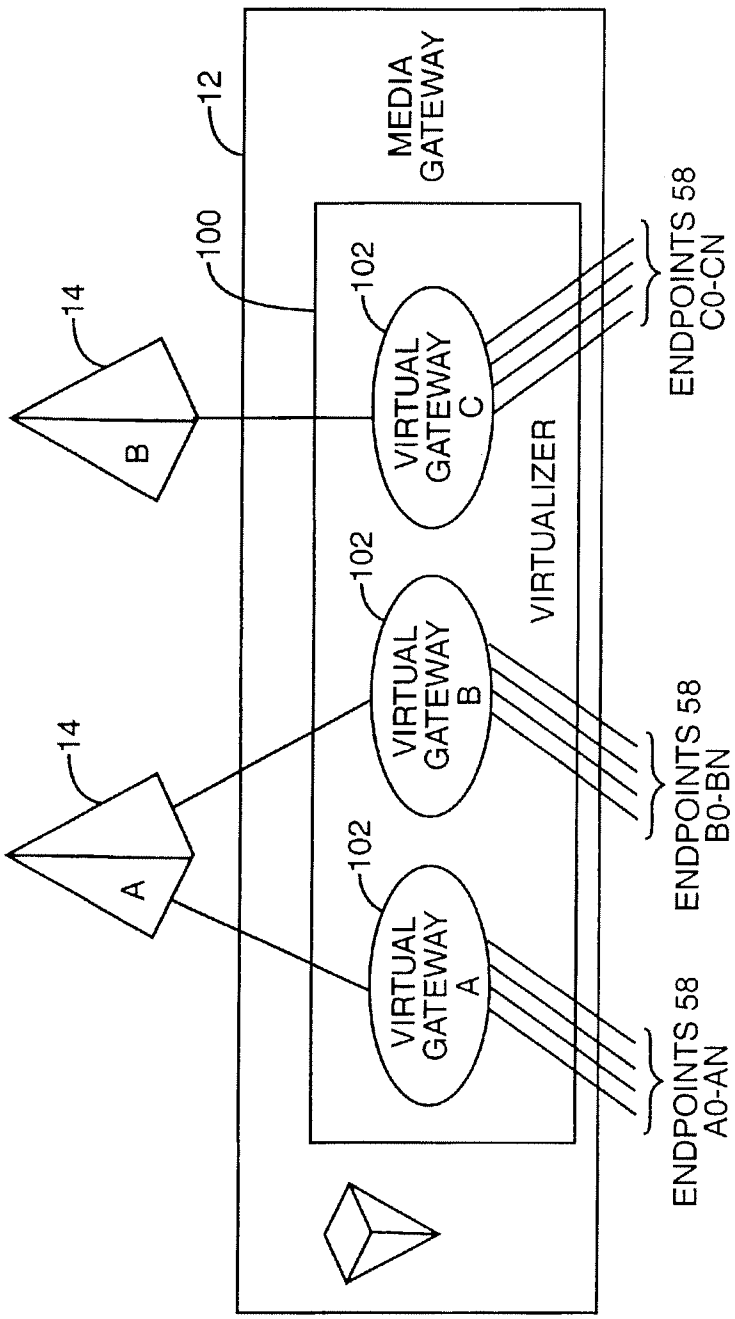

12
MEDIA GATEWAY
100
102
VIRTUAL GATEWAY C
VIRTUAL GATEWAY B
VIRTUAL GATEWAY A
VIRTUALIZER
14
B
A
ENDPOINTS 58 C0-CN
ENDPOINTS 58 B0-BN
ENDPOINTS 58 A0-AN

FIG. 6

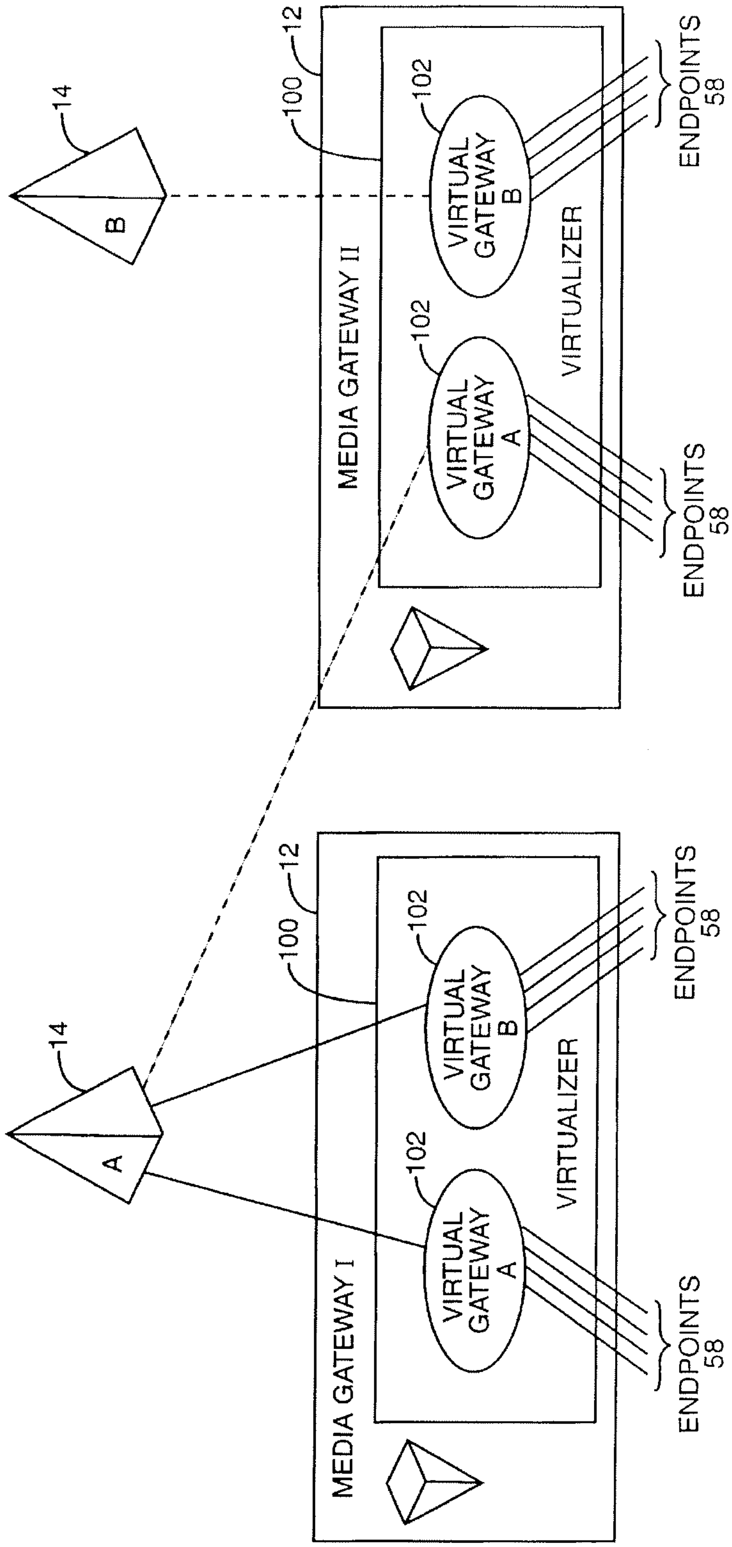
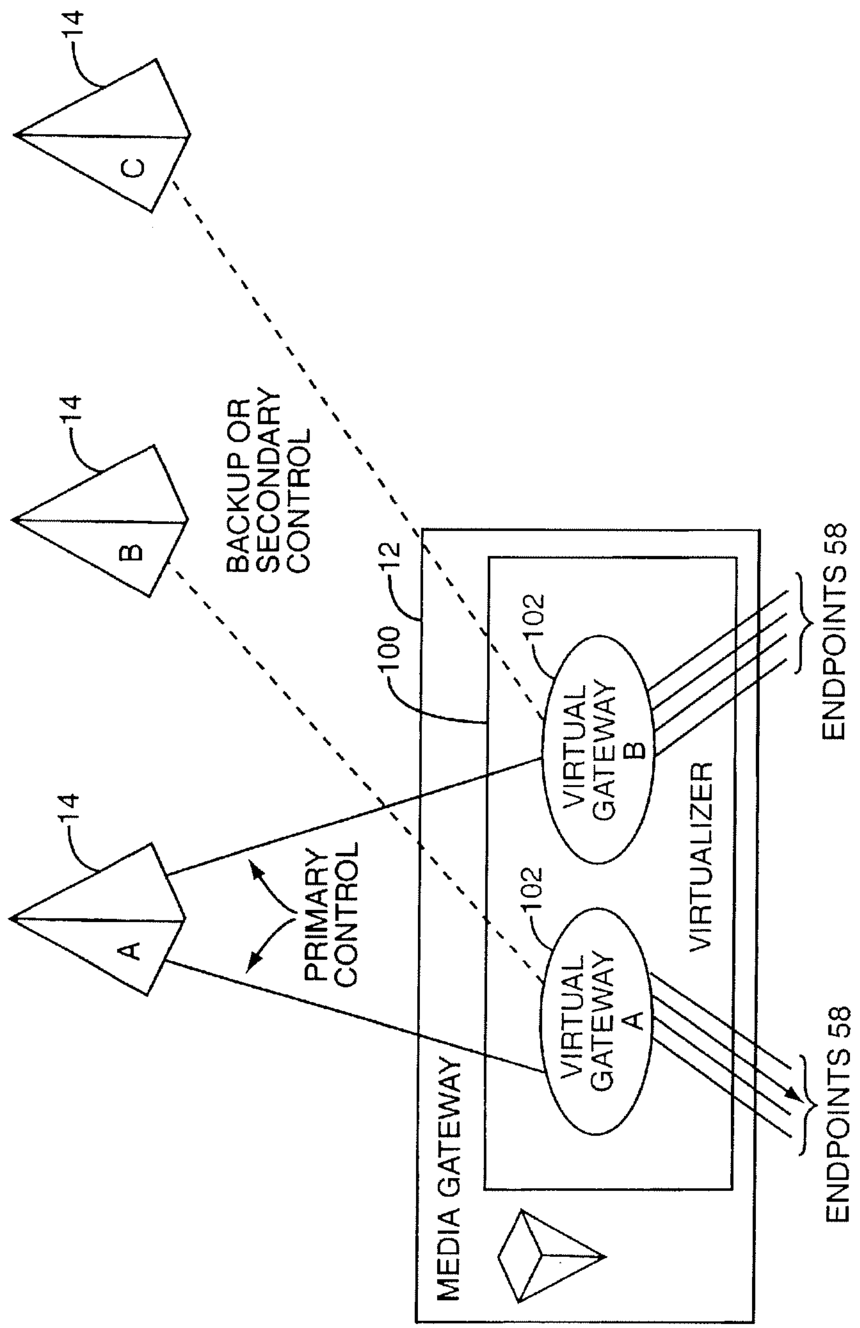

14
B
100
12
MEDIA GATEWAY II
102
VIRTUAL GATEWAY B
102
VIRTUAL GATEWAY A
VIRTUALIZER
ENDPOINTS 58
ENDPOINTS 58
14
A
100
12
MEDIA GATEWAY I
102
VIRTUAL GATEWAY B
102
VIRTUAL GATEWAY A
VIRTUALIZER
ENDPOINTS 58
ENDPOINTS 58

FIG. 7



14
C
14
B
BACKUP OR SECONDARY CONTROL
14
A
PRIMARY CONTROL
12
100
102
VIRTUAL GATEWAY B
102
VIRTUAL GATEWAY A
VIRTUALIZER
MEDIA GATEWAY
ENDPOINTS 58
ENDPOINTS 58

FIG. 8

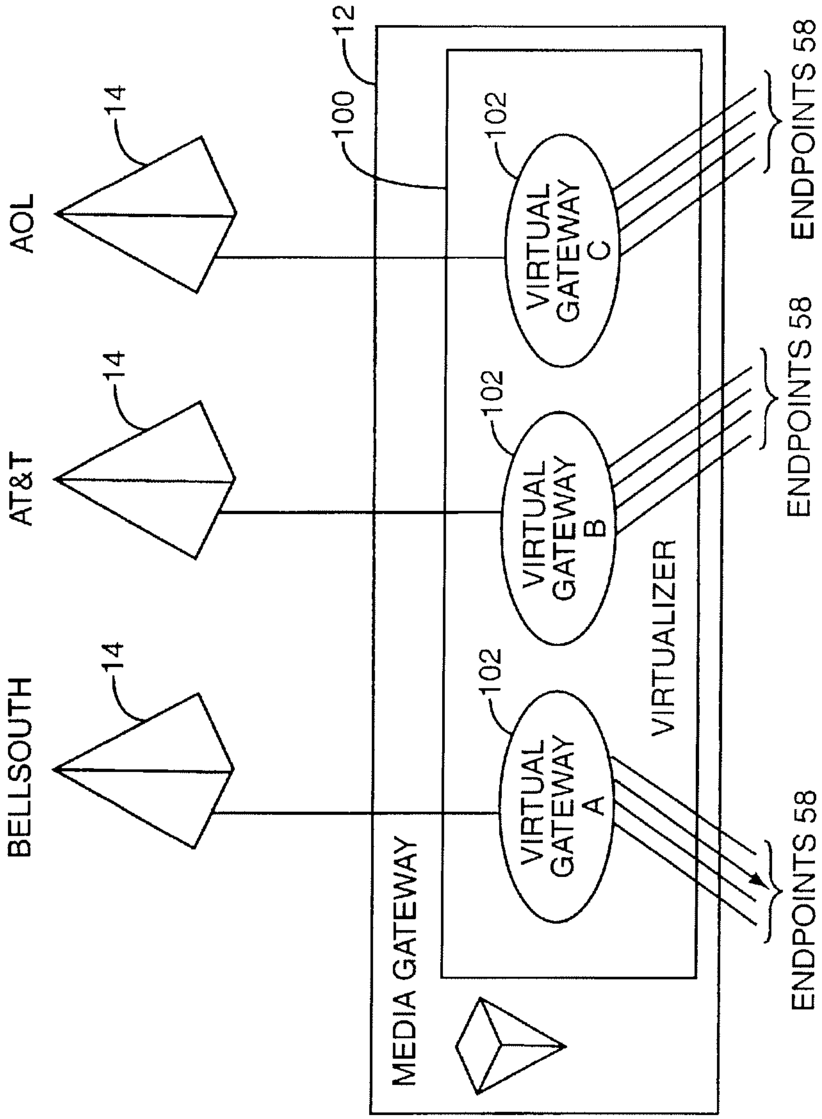

AOL
AT&T
BELLSOUTH
14
14
14
12
100
102
102
102
VIRTUAL GATEWAY C
VIRTUAL GATEWAY B
VIRTUAL GATEWAY A
VIRTUALIZER
MEDIA GATEWAY
ENDPOINTS 58
ENDPOINTS 58
ENDPOINTS 58

FIG. 9

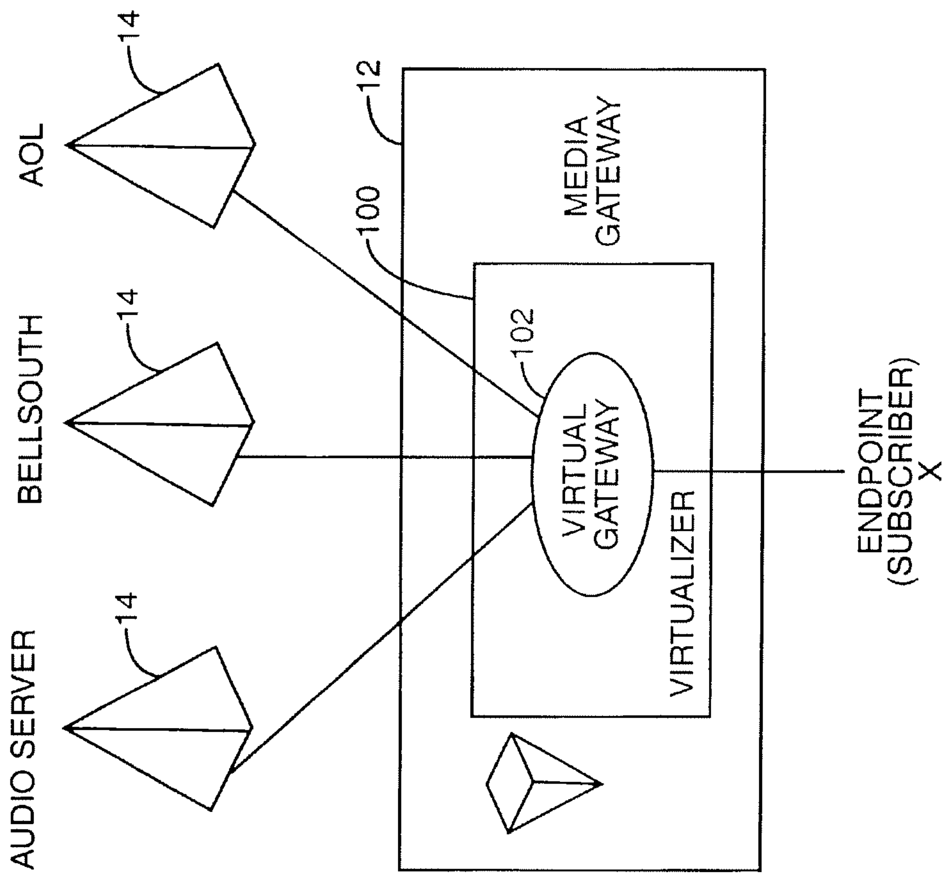

AOL
BELLSOUTH
AUDIO SERVER
14
14
14
12
MEDIA GATEWAY
100
102
VIRTUAL GATEWAY
VIRTUALIZER
ENDPOINT (SUBSCRIBER) X

FIG. 10

DYNAMIC ASSOCIATION OF ENDPOINTS TO MEDIA GATEWAY CONTROLLERS

FIELD OF THE INVENTION

The present invention relates to telecommunications, and particularly, to providing telephony services from disparate media gateway controllers for a single media gateway.

BACKGROUND OF THE INVENTION

There is a growing interest in the convergence of the public switched telephone network (PSTN), the Internet and other internets and intranets. The convergence of these networks requires technology that facilitates interworking in a uniform and effective manner. The next generation of unified networks will provide an open and scalable architecture to accommodate multiple vendors and protocols under a single communication scheme. At the moment, there are several obstacles to providing a scalable, unified network incorporating the PSTN, Internet, cable systems, and wireless systems, among other existing and further networks.

The traditional PSTN provides constant bandwidth streams of information between users. These media streams travel over dedicated circuits, typically between telephones. Circuit-switched networks were originally designed for carrying voice traffic and handling calling patterns, but, with the emergence of the Internet, are now handling significant amounts of data traffic. The data traffic occupies a significant amount of the bandwidth of the circuit-switched network as the data makes its way to Internet protocol (IP)-based networks. In addition, the IP-based networks are now carrying significant amounts of data that relate to voice, fax and video in addition to conventional data. Further, advances in cable and wireless technologies are requiring cable networks and wireless networks to efficiently interact with the PSTN and the IP-based Internet.

Since packet switching networks appear to be the common thread between all of the many networks, there is a need to seamlessly interwork all networks and individual endpoints connecting to these networks. The interface between networks, as well as between a network and individual endpoints, is provided by media gateways. Media gateways require interaction with media gateway controllers to provide decision-making and coordination with other media gateways.

The primary responsibility of the media gateway is to allow media of various types, including voice, fax, video and data to be transported in a unified network. Typically, the media must be transportable both as packets in an IP-based network and as digital or analog streams in a circuit-switched network. In such applications, the media gateway provides bi-directional communications between a circuit-switched network and media-related elements associated with an IP network. Media gateways generally interact with end users in telephony applications or with other media gateways to facilitate such applications. The media gateway controllers provide media gateways with instructions on interconnecting two or more telephony or IP elements in order to exchange information. For example, media gateway controllers instruct media gateways on how to set up, handle and terminate media flows, such as Internet connections or telephone calls.

Existing media gateways are very rigid in structure and configuration. Typically, all endpoints associated with a media gateway are served by a single media gateway controller. Although the H.248 protocol standard for packet telephony set forth by the International Telecommunications Union and the Internet Engineering Task Force indicates that groups of endpoints within a single gateway may be handled by separate media gateway controllers, there is a need for increasing the flexibility of handling endpoints by multiple media gateway controllers and providing an efficient way of allocating endpoints with one or more media gateway controllers. Given the present standards and architectures, it is likely a given media gateway will have only a small number of defined groupings of endpoints capable of being served by different media gateway controllers. Thus, a single media gateway will likely be controlled by only a very small number of media gateway controllers.

In order to maximize the service and selection to end users, it is desirable to enable individual lines or endpoints of a media gateway to be supported by any media gateway controller throughout the network. Current architectures do not allow subscribers at any given endpoint to receive media and data services provided by media gateway controllers operated by different service providers, nor do they allow media gateways to interact with call servers supporting variations in communication and media control protocols.

The existing rigidity in media gateways prevents 1) dynamically changing associations between endpoints and media gateway controllers to facilitate efficient changes in service for users in a given endpoint; 2) easily scaling the system to accommodate additional endpoints; and 3) readily reassigning and distributing endpoints to other media gateway controllers in case of congestion or failure of a media gateway controller.

As such, there is a need for a more efficient and flexible architecture allowing dynamic association of media gateway endpoints with any number of media gateway controllers.

SUMMARY OF THE INVENTION

The present invention addresses the failings of the state-of-the-art by providing an architecture for a media gateway to identify and register with multiple media gateway controllers for various types of voice and data services, along with having the media gateway appear to each of these media gateway controllers as a single, dedicated control entity. A logical layer, referred to as a virtualizer, is associated with each media gateway and appears to the media gateway as a single media gateway controller. To each media gateway controller the virtualizer supports, the virtualizer appears as a single media gateway. In essence, the virtualizer is a protocol manager and message router. The virtualizer supports the registration of multiple media gateways and then creates virtual gateways based on requirements of the media gateway controllers serving these virtual gateways. Preferably, subscribers associated with endpoints handled by the media gateway are grouped into a virtual gateway being served by a select media gateway controller or group thereof.

The virtualizer interacts with the network manager to determine the media gateway controllers it should register against, depending on the type of services requested by the subscribers. As events are reported from the media gateway, the virtualizer routes resulting messages to the appropriate media gateway controller. Conversely, the media gateway commands received from the media gateway controller are forwarded to the appropriate media gateway. The virtualizer may translate commands or event reports as required in the event the media gateway or media gateway controller are not using identical versions of a particular protocol.

The virtualizer and the virtual gateways associated therewith may be incorporated in a media gateway, a media

gateway controller, or a third device coupled to an accessible network. Importantly, the system handling the virtualizer operation is configured to dynamically associate endpoints for one or more gateways as a virtual gateway and arrange the necessary relationship with one or more media gateway controllers. The dynamic handling and creation of these relationships allows efficient controlling of individual lines and endpoints to be supported from any media gateway controller in the network. Further, subscribers at any given endpoint may receive media and data services from multiple service providers at any given time. The ability to dynamically change associations allows for efficient scaling of the system to add media gateways and endpoints to existing, as well as new, media gateway controllers. Further, the virtualizer may support a secondary or backup association with a media gateway controller in case the primary media gateway controller fails or becomes overly congested.

Other aspects and features of the present invention will become apparent to those ordinarily skilled in the art upon reviewing the following description of the preferred embodiments of the invention in conjunction with the accompanying figures.

BRIEF DESCRIPTION OF THE DRAWINGS FIGURES

FIG. **1** is an overview of a comprehensive network incorporating multiple applications of the present invention.

FIG. **2** is a block representation of a media gateway according to the present invention.

FIG. **3** is a block schematic of a media gateway controller according to the present invention.

FIG. **4** is a block representation of the interaction between media gateways and media gateway controllers according to the present invention.

FIG. **5** is a block representation of a virtualizer logic layer according to the present invention.

FIG. **5A** is a chart providing an example mapping between endpoints on multiple gateways and media gateway controllers.

FIG. **6** is a block representation of a virtualizer providing multiple virtual gateways within a single media gateway.

FIG. **7** is a block representation of scaling endpoint allocation.

FIG. **8** is a block representation outlining the reallocation of a media gateway controller when a primary media gateway controller fails or requires reassignment due to traffic.

FIG. **9** is a block representation of a media gateway apportioning endpoints among multiple media gateway controllers capable of providing media control as well as services.

FIG. **10** is a block representation of a media gateway allowing access from multiple media gateway controllers and service providers to a single endpoint or subscriber.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The focus of the present invention is to facilitate seamless interworking of communication networks, including the public switched telephone network (PSTN), SS7 networks, cable, wireless, and multi-purpose packet switching networks, such as IP-based networks. To facilitate the interconnection and interworking of various ones of these networks, there are two primary elements: a media gateway and a media gateway controller. The media gateway provides the actual interface between networks and connection of various subscribers via endpoints. The media gateway controller provides decision-making and coordination between media gateways to facilitate interworking.

The primary responsibility of a media gateway is to allow media of various types, such as data, voice and video, to be transported in unified networks. Media gateways are configured to allow media to be transportable, both as packets in an IP or ATM network and as analog or digital streams in more traditional circuit-switched networks. The gateways allow media to move without loss of integrity or quality between networks and network technologies. In essence, the media gateway provides a bi-directional interface between networks, and typically, between a circuit-switched network and media-related elements in an IP network. Media gateways may interact with IP telephony applications residing in computers attached to a network, or with other media gateways. Media gateways may also provide physical interfaces to the public switched telephone network and may be used to replace switching components normally used in the plain old telephone system (POTS).

The primary responsibilities of the media gateway controller are to make decisions based on flow-related information and to provide instructions on interconnecting elements or endpoints within and throughout the networks. Media gateway controllers store status information on media flows and may be used to generate administrative records for a variety of media-related activities, such as billing. Most prominently, media gateway controllers provide coordination of media gateways. Typically, media gateway controllers direct media gateways to set up, handle and terminate individual media flows. As an analogy, media gateway controllers would implement the call control functionality found in switching elements in a PSTN. The switching elements that provide the media path for the call are analogous to the media gateways. Notably, this disclosure anticipates the replacement of switching elements of the PSTN with media gateways and media gateway controllers.

FIG. **1** provides an outline of the various types of configurations and implementations for media gateways and media gateway controllers. As depicted, an ATM and/or IP packet network **10** may be the center of a particular network topology. The packet network **10** may represent a single network or thousands of networks operating together to form an internet. Hanging off of the packet network **10** are a variety of elements capable of communicating with one another and elements on additional networks. With particular reference to the present invention, a series of media gateways **12A-12H** are shown coupled to a variety of systems and networks. For example, media gateways **12A**, **12B** and **12G** are configured to connect to and cooperate with traditional telephony systems, such as the carrier office **20**, which serves a municipality or community **18**. In this instance, the media gateway **12A** provides a trunk interface to the carrier office **20** from the packet network **10**.

Similarly, media gateway **12B** provides a trunk and individual line interfaces to a switch or end office **22** as well as to a switch or tandem **24**. Media gateway **12B** also provides individual endpoints for providing local telephone loops for telephones **26**. The systems may also be configured to handle private branch exchanges and the like.

Media gateway **12G** provides line interfaces to various points throughout a community **50** to provide any number of telephony services such as POTS, ISDN, DSL, voice over IP, DS-**1**, DS-**3**, or other private line services. Importantly, these examples are provided to show that the media gateway

may provide a variety of interfaces to the traditional telephone system and replace analogous elements therein. In these configurations, the gateway provides a circuit-switched, time-division multiplexing (TDM) interface to the traditional telephony network and a packet-switched interface to the packet network **10**.

Continuing with FIG. **1**, another exemplary gateway configuration provides a cable interface. Media gateway **12**D provides the interface between the packet network **10** to a residence **30** to provide data, voice and audio to any number of devices, including televisions **32** and telephones **34**.

Gateway **12**E is configured to couple the packet network **10** to a wireless network **36**. A wireless interface allows communications from any number of mobile terminals, such as wireless telephones **40**, vehicles **42**, and modems or wireless local area networks for residences or businesses **44**. The media gateway **12**E may function as a base station or mobile switching center.

Continuing with the possible options, media gateway **12**F is configured to be an Internet service provider media gateway that couples to an ISP or intranet of ISPs **46**. As noted, media gateways may be configured as media gateway **12**H, wherein an interface is created between the ATM or IP packet network **10** and another packet network **52** using a different networking technology or protocol.

In addition to the media gateways, the media gateway controllers **14**A–**14**E are coupled to the packet network **10** to provide control for the media gateways as well as control services to the gateway or users connected to the gateway. As noted, the media gateway controllers may instruct media gateways on how to set up, handle and terminate individual media paths and flows between media gateways. These functions are discussed in greater detail below. The media gateway controllers **14** may also cooperate with or function as media servers to provide data, audio and video content. Further, as shown in association with media gateway controller **14**C, the media gateway controller may provide an interface for an SS7 network **16**, which is used to set up, handle and terminate calls over the PSTN.

As is apparent from FIG. **1**, the media gateway controllers **14** and media gateways **12** facilitate the convergence of differing networks to provide integrated interworking systems that bring together traditional telephony with next generation communication networks and systems.

Referring now to FIG. **2**, a block schematic of a media gateway **12** is shown having a central processing unit (CPU) **110** containing memory **112** and the requisite software **114**. The CPU **110** cooperates to provide a bi-directional interface between a packet interface **116** and the media interface **118**. The packet interface **116** would preferably connect to a packet network, as referenced in FIG. **1**, while the media interface **118** is the interface opposite the packet network **10** and may range from traditional trunks and lines of a telephone network to a cable or wireless system. As such, the media interface **118** may handle various packet or circuit-switching technologies. For example, when interfacing with traditional telephony, the media interface **118** is configured to handle TDM communications or any other analog or digital data streams required to facilitate communications. If used to interface with another packet network, the media interface can handle packet-switching technologies. Regardless of configuration, the media interface **118** is typically associated with multiple addresses or locations for end users or end applications. For the purpose of the discussion herein and the claims that follow, the term "endpoints" is used to correspond to a particular address, designated application, or telephone number.

FIG. **3** is a block schematic of a media gateway controller **14** having a CPU **120** with associated memory **122** and software **124**. The media gateway controller **14** will have at least one interface, preferably a packet interface **126** capable of communicating across the ATM/IP packet network **10**.

For the purpose of describing the preferred embodiments of the present invention, the following description assumes the media gateways **12** provide a bi-directional interface between a circuit-switched network, such as the TDM-based public switched telephone network and media-related elements in an IP network. It is also assumed that the media gateways will interact either with IP telephony end-user applications residing in computers attached to the IP network, with end-user telephones, or with other media gateways. It is important to understand that the media gateways can implement a variety of physical interfaces to the PSTN. For example, the media gateway may implement a high-speed TDM trunk interface or line interface, which are commonly used interfaces between switching elements in a circuit-switched network. Thus, as shown in FIG. **4**, media gateways **12** provide an interface to a plurality of endpoints **58**, which correspond with IP addresses or phone numbers in a TDM network. On the other side of the media gateway **12**, the packet interface will interface with the IP/ATM packet network **10** to communicate with one or more media gateway controllers **14** and other media gateways **12**.

Conventional media gateways are configured to associate one or more sets of endpoints to one or more media gateway controllers **14**. Prior to Applicant's invention, associating endpoints of a media gateway with a specific media gateway controller was cumbersome and inefficient. Currently, there is no way to easily define these associations or change such associations dynamically. When establishing a call or media path **60** between an originating endpoint **58** on one media gateway **12** and a terminating endpoint **58** on another media gateway **12**, the respective media gateways **12** would contact their associated media gateway controllers **14**, which cooperate with one another and with the media gateways to establish the media path **60**. The rather inflexible nature of media gateways and their association of endpoints with media gateway controllers results in inefficiencies when new endpoints are added. Further, there is no way to efficiently reassign a media gateway controller to endpoints when a media gateway fails or when increases in the number of endpoints and traffic requires reassignment to increase efficiency of the network.

In contrast to these traditional media gateway configurations and as shown in FIG. **5**, the present invention provides a logical layer, referred to as a virtualizer **100** between a media gateway **12** and a media gateway controller **14**. The virtualizer **100** is configured to dynamically create associations between media gateway controllers **14** and a set of physical endpoints **58** on one or more media gateways **12**. The virtualizer **100** allows for the creation of virtual gateways **102** that effect a media gateway controller interface to media gateways **12**, and a media gateway interface to media gateway controllers **14**. The virtualizer **100** may also provide a service broker **104**, which interacts with network directories through a network manager **54** to determine how endpoints **58** should be distributed between and among the media gateway controllers **14**.

For example, the representation of the virtualizer **100** in FIG. **5** may allocate endpoints on a number of media

gateways **12** to media gateway controllers **14** as shown in FIG. **5**A. Notably, the virtualizer **100** provides three virtual gateways **102** that are respectively labeled A, B and C. Virtual gateway A (**102**) maps the endpoints **W0** through WN of media gateway W (**12**) and endpoints X1 through X3 of media gateway X (**12**) to media gateway controller X (**14**). Virtual gateway B (**102**) maps the remaining endpoints of media gateway X (**12**), X4 through XN, to media gateway controller Y (**14**). Virtual gateway C (**102**) maps all the endpoints (**58**) from media gateways Y (**12**) and Z (**12**) to media gateway controller Z (**14**). The service broker **104** may be used for initial setup and configuration of the virtualizer **100** to distribute the endpoints **58** for the various media gateways **12** to the designated media gateway controllers **14**.

The virtualizer **100** allows each virtual gateway **102** to appear to the physical media gateway as a single media gateway controller. To each media gateway controller the virtualizer supports, the virtual gateways appear as a single media gateway. Preferably, the virtualizer **100** is a protocol manager and message router. The virtualizer **100** supports the registration of multiple gateways **12** and then creates virtual gateways **102** based on the media gateway controller requirements of the entire set of endpoints. Each virtual gateway **102** is then registered with its controlling media gateway controller as a separate gateway.

The virtualizer **100** may interact with the network manager **54** to determine which media gateway controllers **14** it should register against, depending on the type of services requested by the subscriber at any of the given endpoints. As events are reported from the physical, media gateway **12**, the virtualizer **100** simply routes the message to the appropriate media gateway controller **14**. Conversely, media gateway commands received from the media gateway controller **14** are forwarded to the appropriate media gateway **12**. Optionally, the virtualizer **100** may translate the command or event report in the event that the media gateway controller **14** and the media gateway **12** are not using identical versions of a protocol or different protocols altogether.

Importantly, the virtualizer **100** and the virtual gateways **102** provided thereby can be implemented in the media gateway **12**, the media gateway controller **14**, a separate network entity, or combination thereof. Further, the virtualizer **100** and associated virtual gateways **102** may be isolated to one of these devices or spread over any number of these devices. For the remaining portion of the detailed description, it is assumed that the virtualizer **100** is run on a single media gateway **12** as shown in FIG. **6**.

FIG. **6** depicts a media gateway **12** having a virtualizer **100** and three virtual gateways **102**, which are designated A, B and C. All the endpoints A**0** through AN, B**0** through BN and C**0** through CN are allocated to respective virtual gateways A, B and C. Endpoints A**0** through AN are associated with media gateway controller A via virtual gateway A. Similarly, endpoints B**0** through BN are associated with media gateway controller A via virtual gateway B. In contrast, endpoints C**0** through CN are associated with media gateway controller B via virtual gateway C. The virtualizer may interact with a variety of network elements, such as the network manager **54** via a service broker **104** to dynamically change the association of endpoints with the media gateway controllers **14**. For example, endpoints A**0** through AN or any portion thereof may be dynamically reallocated to media gateway controller B in real time without the need for the timely configuration associated with current media gateways.

For registration, media gateways **12** preferably interact with an element manager **56** (FIG. **1**) to determine the IP address for their primary media gateway controller **14**. The primary media gateway controller's IP address will map to the virtualizer **100**, and in particular, to a specific virtual gateway **102** in the virtualizer **100**.

If a service change or end-point activation message, such as an H.248 service message, is sent to the virtualizer **100**, it will interact with the service broker **104**, and the network manager **54** (FIG. **1**) if necessary, to identify the media gateway controllers **14** associated with the endpoints **58** identified in the service change request. A virtual gateway **102** is created for the identified group of endpoints **58** and a signaling link is established between the virtual gateway **102** and the corresponding media gateway controller **14**. The service change message is then sent to the media gateway controller **14**. The virtualizer **100** maintains a mapping between endpoints **58**, the virtual gateway **102**, and the physical media gateway **12**. Responses from the media gateway controller **14** are sent to the virtual gateway **102**. The virtual gateway **102** maps this to a physical gateway **12** and forwards the message to the physical gateway **12**.

For media or call control, call control messages, such as H.**248** control messages, are sent from the media gateway **12** to the virtualizer **100**. The virtualizer **100** maps the endpoints **58** to a virtual gateway **102**, and then passes the message via a signaling point associated with the virtual gateway **102** to the proper media gateway controller **14**. Responses from the media gateway controller **14** are routed in reverse direction to the media gateway **12**. In essence, the virtualizer **100** performs a message routing function that is dynamically configurable.

In addition to the ready ability to re-configure and re-register endpoints for a media gateway **12** with a combination of media gateway controllers **14**, the virtualizer **100** provides easy growth and scaling for media gateway controllers **14**. With reference to FIG. **7**, media gateway I (**12**) has a virtualizer **100** with two virtual gateways **102** associated with corresponding sets of endpoints **58**. The virtual gateways **102** associated with media gateway I (**12**) are mapped to media gateway controller A (**14**). The solid lines coupling media gateway controller A (**14**) and virtual gateways A and B (**102**) represent an initial configuration. If additional endpoints are added and media gateway controller A (**14**) has excess capacity, a new media gateway II (**12**) may be provided with a virtualizer **100** having two virtual gateways **102**. In the present example, virtual gateway A of media gateway II is mapped to media gateway controller A (**14**) while virtual gateway B (**102**) is mapped to media gateway controller B (**14**). In essence, the system enables distribution of endpoints of a large gateway between multiple media gateway controllers as well as distribution of media gateway controllers among multiple media gateways. These distributions are dynamically configurable in real time using internal controls or instructions from any number of network devices, including media gateway controllers.

The scaling ability provided by the present invention is particularly useful when the capacity of a trunk or business group must be increased. Any additional capacity for the particular trunk or business group may be handled by another media gateway and the same media gateway controller as the original group.

Referring now to FIG. **8**, another benefit provided by dynamic allocation of endpoints among multiple media gateway controllers is the ability to maintain the mapping of the endpoints to both primary and backup media gateway controllers in order to protect against loss of a media gateway controller. If a failure occurs, the virtual gateways

A and B (**102**) within the virtualizer **100** are re-assigned or re-registered as a unit from media gateway controller A (**14**) to one or more backup media gateway controllers B and C (**14**). Such an ability is very useful in the management of groups of endpoints **58** that cannot be distributed between media gateway controllers **14**, such as members of trunk groups or business groups. Preferably, the mapping tables maintained in the virtualizer **100** are replicated in a backup node to protect against loss of the virtualizer itself. A backup virtualizer would re-register each virtual gateway **102** with another media gateway controller **14** in the event that the primary virtualizer **100** is no longer available. The re-registration would preferably work as discussed above using the service broker **104** in cooperation with the network manager **54**.

Partial allocation of endpoints **58** within a single media gateway **12** between multiple media gateway controllers **14** allows specific groups of endpoints **58** to interact with any number of service providers associated with a particular media gateway controller **14**. As shown in FIG. **9**, a single media gateway **12** may have three virtual gateways **102** within the virtualizer **100** servicing three sets of endpoints **58**. Each set of endpoints **58** is registered with and receives service from different service providers, such as BellSouth, AT&T and America Online (AOL). Notably, these services may be anything from Internet connectivity to traditional telephony services. Importantly, endpoints **58** associated with each of the provider's media gateway controllers **14** may be grouped dynamically into separate virtual gateways **102**.

In addition to allocating groups of endpoints **58** to particular service providers, any single endpoint **58** may be associated with multiple service providers as shown in FIG. **10**. For example, an individual subscriber at a particular endpoint X may desire to initiate service requests to various content providers. A virtual gateway **102** may facilitate signaling connections to multiple providers and integrate the media stream provided to a user according to the user's equipment capability and configuration. In the example of FIG. **10**, the individual may have an audio server, such as an MP3 provider, a telephony service, such as BellSouth, and an Internet service, such as AOL. Notably, the virtual gateway **102** may dynamically add additional service providers by registering the media gateway **12** with the requisite media gateway controller **14** or service content provider as desired by the subscriber.

Those of ordinary skill in the art will recognize modifications to the invention based on the teachings above. All such modifications are considered within the scope of the invention and the claims that follow.

What is claimed is:

1. A system for associating endpoints of a media gateway with a plurality of media gateway controllers, said system comprising a control system adapted to dynamically associate at least one endpoint from a media gateway such that said at least one endpoint can receive service from any one of a plurality of media gateway controllers within a network independent of a second media gateway controller providing service to other endpoints from the media gateway wherein said control system has a logical layer providing virtual gateways for mapping said at least one endpoint to a media gateway controller, such that each said virtual gateway appears as a media gateway to a media gateway controller and appears as a media gateway controller to a media gateway.

2. The system of claim **1** wherein said control system is further adapted to dynamically associate at least one end-

point with a virtual gateway and register said virtual gateway with a corresponding one of the media gateway controllers.

3. The system of claim **1** wherein said control system is flintier adapted to dynamically associate at least one endpoint to be registered with a certain media gateway controller and create one said virtual gateway mapping said at least one endpoint to the media gateway controller.

4. The system of claim **1** wherein said control system is further adapted to communicate with a network manager to identify a select one of the media gateway controllers to associate with said at least one endpoint and configure the associated virtual gateway accordingly.

5. The system of claim **1** wherein said control system is filer adapted to cooperate with said logical layer to provide protocol translation between a first protocol used by the media gateway and a second protocol used by the media gateway controller.

6. The system of claim **1** wherein said control system is further adapted to cooperate with said logical layer to reassign an endpoint from one virtual gateway to another virtual gateway.

7. The system of claim **1** wherein said control system is further adapted to cooperate with said logical layer to assign an endpoint to an existing virtual gateway being served by a media gateway controller.

8. The system of claim **1** wherein said control system is further adapted to cooperate with said logical layer to store a secondary media gateway controller for serving a virtual gateway and reassign the virtual gateway from a primary media gateway controller to the secondary media gateway controller as conditions necessitate.

9. The system of claim **8** wherein said control system is further adapted to cooperate with said logical layer to register said reassigned virtual gateway with said secondary media gateway controller.

10. The system of claim **1** wherein said control system is further adapted to cooperate with said logical layer to associate a first virtual gateway to a first media gateway controller for a first service provider and a second virtual gateway to a second media gateway controller for a second service provider such that endpoints on a common media gateway may be served by multiple service providers.

11. The system of claim **1** wherein said control system is further adapted to dynamically associate one endpoint from a media gateway with a plurality of media gateway controllers such that said one endpoint can receive service from said plurality of media gateway controllers.

12. The system of claim **1** wherein said control system is further adapted to cooperate with said logical layer to associate select endpoints from a plurality of media gateways with one virtual gateway.

13. The system of claim **1** wherein said control system is embedded within a media gateway.

14. The system of claim **1** wherein said control system is embedded within a media gateway controller.

15. The system of claim **1** wherein said control system is embedded within a device other than a media gateway and media gateway controller.

16. The system of claim **1** wherein said control system facilitates a service broker function adapted to assist in identifying at least one media gateway controller for the at least one endpoint.

17. A computer readable media comprising software for instructing a control system to dynamically associate at least one endpoint from a media gateway such that said at least one endpoint can receive service from any one of a plurality

of media gateway controllers within a network independent of a second media gateway controller providing service to other endpoints from the media gateway wherein endpoints of a media gateway may be associated with different media gateway controllers wherein said control system is further instructed to create a logical layer providing virtual gateways for mapping said at least one endpoint to a media gateway controller, such that each said virtual gateway appears as a media gateway to a media gateway controller and appears as a media gateway controller to a media gateway.

18. The computer readable media of claim **17** wherein said control system is further instructed to dynamically associate at least one endpoint with a virtual gateway and register said virtual gateway with a corresponding one of the media gateway controllers.

19. The computer readable media of claim **17** wherein said control system is further instructed to dynamically associate at least one endpoint to be registered with a certain media gateway controller and create one said virtual gateway mapping said at least one endpoint to the media gateway controller.

20. The computer readable media of claim **17** wherein said control system is further instructed to cooperate with said logical layer to store a secondary media gateway controller for serving a virtual gateway and reassign the virtual gateway from a primary media gateway controller to the secondary media gateway controller as conditions necessitate.

21. The computer readable media of claim **20** wherein said control system is further instructed to cooperate with said logical layer to register said reassigned virtual gateway with said secondary media gateway controller.

22. The computer readable media of claim **17** wherein said control system is further instructed to cooperate with said logical layer to associate a first virtual gateway to a first media gateway controller for a fist service provider and a second virtual gateway to a second media gateway controller for a second service provider such, that endpoints on a common media gateway may be served by multiple service providers.

23. The computer readable media of claim **17** wherein said control system is further instructed to dynamically associate one endpoint from a media gateway with a plurality of media gateway controllers such that said one endpoint can receive service from said a plurality of media gateway controllers.

24. The computer readable media of claim **17** wherein said control system is further instructed to cooperate with said logical layer to associate select endpoints from a plurality of media gateways with one virtual gateway.

25. A system for associating endpoints of a media gateway with a plurality of media gateway controllers, said system comprising means adapted to dynamically associate at least one endpoint from media gateway such that said at least one endpoint can receive service from one of a plurality of media gateway controllers within a network independent of a second media gateway controller providing service to other endpoints from the media gateway, said means adapted to dynamically associate further provides virtual gateways for mapping said at least one endpoint receiving service to a media gateway controller, such that each said virtual gateway appears as a media gateway to a media gateway controller and appears as a media gateway controller to a media gateway.

26. The system of claim **25** further comprising:

a) means to interface with a plurality endpoints of a media gateway;

b) means to interface with a packet network to facilitate communication with said media gateway controller, and

c) means to facilitate a media path to after media entity based on instruction from the media gateway controller.

27. A method of associating endpoints of a media gateway with a plurality of media gateway controllers comprising:

a) providing virtual gateways associated with a media gateway;

b) associating at least one endpoint of at least one media gateway with one said virtual gateway such that said at least one endpoint can receive service from one of a plurality of media gateway controllers via the virtual gateway independent of a media gateway controller providing service to other endpoints from the media gateway;

c) mapping signals received from the media gateway controller to the media gateway for the at least one endpoint; and

d) mapping signals received from the media gateway for the at least one endpoint to the media gateway controller.

28. The method of claim **27** wherein providing virtual gateways associated with the media gateway comprises providing virtual gateways associated with the media gateway with a logical layer.

* * * * *

CERTIFICATE OF SERVICE

The undersigned hereby certifies that on December 14, 2007, a true and correct copy of the foregoing **Nortel's Answer and Counterclaims** was served upon the following counsel of record as indicated:

VIA HAND DELIVERY

Steven J. Balick, Esquire
John G. Day, Esquire
Tiffany Geyer Lydon, Esquire
Ashby & Geddes
500 Delaware Avenue, 8th Fl.
Wilmington, DE 19801

/s/ Denise Seastone Kraft
Denise Seastone Kraft (#2778)
Edwards Angell Palmer & Dodge LLP
Wilmington, DE 19801
(302) 777-7770
dkraft@eapdlaw.com